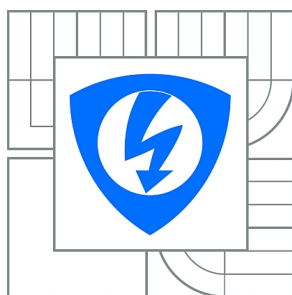




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ**

ÚSTAV RADIOELEKTRONIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF RADIO ELECTRONICS

INTERFACE DIAGNOSTICKÉ SBĚRNICE PRO AKUMULÁTORY

ACCUMULATOR DIAGNOSTIC BUS INTERFACE

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. FRANTIŠEK CHYTKA

VEDOUcí PRÁCE

SUPERVISOR

doc. Ing. JAROMÍR KOLOUCH, CSc.

BRNO 2013



**VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ**

**Fakulta elektrotechniky
a komunikačních technologií**

Ústav radioelektroniky

Diplomová práce

magisterský navazující studijní obor
Elektronika a sdělovací technika

Student: Bc. František Chytka

ID: 115188

Ročník: 2

Akademický rok: 2012/2013

NÁZEV TÉMATU:

Interface diagnostické sběrnice pro akumulátory

POKYNY PRO VYPRACOVÁNÍ:

Prostudujte principy diagnostiky akumulátorů používané pro monitorování stavu akumulátorů v notebookech a podobných zařízeních s diagnostickou sběrnicí Smart Battery s uvažováním standardů SMBus a I2C. Navrhněte koncepci monitorovacího zařízení pro baterie s různými komunikačními čipy, které bude předávat potřebné údaje do řídicího počítače PC.

Vypracujte podrobný návrh obvodového řešení monitorovacího zařízení a základní verzi příslušného programového vybavení dovolující posoudit funkčnost monitoru. Navržené řešení realizujte a proveďte potřebné zkoušky k ověření jeho funkčnosti.

Zpracujte konečnou verzi programového vybavení a vytvořte podporu pro existující komunikační čipy včetně vybavení řídicího počítače. Správnou funkci monitoru ověřte čtením údajů na několika notebookových bateriích osazených různými čipy.

DOPORUČENÁ LITERATURA:

[1] KAINKA, B. Měření, řízení a regulace pomocí PC. Praha: BEN - technická literatura, 2003.

[2] Smart Battery Data Specification [online]. – [cit. 8. prosince 2011]. Dostupné na [www: http://sbs-forum.org/specs/sbdat110.pdf](http://sbs-forum.org/specs/sbdat110.pdf)

Termín zadání: 11.2.2013

Termín odevzdání: 24.5.2013

Vedoucí práce: doc. Ing. Jaromír Kolouch, CSc.

Konzultanti diplomové práce: Ing. Ondřej Pavelka, Honeywell

prof. Dr. Ing. Zbyněk Raida

Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Výzkum realizovaný v rámci této diplomové práce byl finančně podpořen projektem
CZ.1.07/2.3.00/20.0007 **Wireless Communication Teams**
operačního programu **Vzdělávání pro konkurenceschopnost**.



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Finanční podpora byla poskytnuta Evropským sociálním fondem
a státním rozpočtem České republiky.

Tento příspěvek vzniknul za podpory projektu CZ.1.07/2.3.00/20.0007 WICOMT,
financovaného z operačního programu Vzdělávání pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

ABSTRAKT

Práce se zabývá návrhem a konstrukcí diagnostického zařízení pro akumulátory. V rámci rozboru jsem popsal a porovnal všeobecně známou sběrnici I²C a jí podobnou sběrnici SMBus, která se vyskytuje v inteligentních notebookových akumulátorech Smart Battery. V návrhu zařízení je použita sběrnice USB, kterou jsem též v rozboru stručně popsal. Navrhl jsem několik možností realizace a provedl nad nimi diskusi. Optimální koncepci jsem pak zrealizoval. Realizace zařízení měla dvě fáze, konstrukci a programování SW. Obě jsou v práci popsány. Nakonec jsou předloženy výsledky testu diagnostického zařízení na několika různých akumulátorech.

KLÍČOVÁ SLOVA

I²C, SMBus, Smart Battery, V-USB, diagnostika akumulátoru, Visual Studio

ABSTRACT

The thesis deals with design and construction of diagnostic device for batteries. In the analysis I described and compared well known I²C bus and quite similar SMBus. The latter bus is used in the laptop batteries called Smart Battery. The design includes the USB bus, which I described briefly in analysis too. I suggested a several designs, discuss them and build the best one of them. The realization of the diagnostic device includes construction aspect and programing aspect. Both aspects are described in the thesis. At last, the results of test diagnostic of several batteries carried out by designed device are introduced.

KEYWORDS

I²C, SMBus, Smart Battery, V-USB, battery diagnostic, Visual Studio

CHYTKA, F. *Interface diagnostické sběrnice pro akumulátory*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav radioelektroniky, 2013. 53 s. Diplomová práce. Vedoucí práce: doc. Ing. Jaromír Kolouch, CSc.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma Interface diagnostické sběrnice pro akumulátory jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne

.....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu diplomové práce doc. ing. Jaromíru Kolouchovi, CSc. a externímu konzultantovi Ing. Ondřeji Pavelkovi za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne

.....

(podpis autora)

OBSAH

Seznam obrázků	vi
Seznam tabulek	viii
Úvod	1
1 Struktura systému	2
1.1 Požadavky na systém	2
1.2 Teoretické možnosti realizace	2
2 Popis dílčích částí	4
2.1 Řídicí jednotka – MCU	4
2.1.1 Požadavky	4
2.1.2 Popis ATmega168	4
2.2 Sběrnice I ² C	5
2.2.1 Vlastnosti sběrnice I ² C	5
2.2.2 Přenos dat	5
2.2.3 Adresování zařízení	7
2.2.4 Arbitráž kolize	7
2.2.5 Použití sběrnice I ² C	7
2.3 Sběrnice SMBus	8
2.3.1 Vlastnosti sběrnice SMBus	8
2.3.2 Časování přenosu dat	9
2.3.3 Protokoly sběrnice SMBus	11
2.3.4 Adresace zařízení	13
2.3.5 Packet Error Code	14
2.3.6 Rozdíly mezi standardy sběrnic SMBus a I ² C	14
2.4 Sběrnice USB	16
2.4.1 Verze sběrnice USB a jejich rozdíly	16
2.4.2 Třídy USB zařízení	16
2.4.3 Signály na sběrnici USB	17
2.4.4 Komunikace na sběrnici USB	17
2.5 Ovládací software	18

2.5.1	V-USB – virtuální USB port na mikrokontroléru.....	18
2.5.2	MS Visual Studio a jeho aplikace.....	19
3	Návrh zařízení	21
3.1	Celkové zapojení.....	21
3.2	Ochranné obvody.....	22
3.2.1	Ochrana proti ESD.....	22
3.2.2	Ochrana proti nadproudu	22
3.3	Napájecí obvod	23
3.4	Návrh plošného spoje.....	24
4	Software systému	25
4.1	Program mikrokontroléru	25
4.1.1	Komunikace na sběrnici SMBus.....	25
4.1.2	Komunikace na sběrnici USB.....	27
4.2	PC aplikace – Smart Battery Monitor.....	29
4.2.1	Připojení a provoz dat na sběrnici USB.....	30
4.2.2	GUI – uživatelské rozhraní	31
5	Testování	34
5.1	Komunikace na sběrnici SMBus.....	34
5.2	Komunikace na sběrnici USB.....	35
5.3	Univerzálnost zařízení	36
5.4	Konektor akumulátorů Smart Battery	37
5.4.1	Akumulátor DELL M911G	37
5.4.2	Příklady některých typů akumulátorů.....	38
	Závěr	40
	Literatura	41
	Seznam symbolů, veličin a zkratk	43
	Seznam příloh	44

SEZNAM OBRÁZKŮ

Obr. 1.1:	Možnosti realizace systému.	3
Obr. 2.1:	Zařízení na sběrnici I ² C [18].	5
Obr. 2.2:	A – Start podmínka, B – stop podmínka, převzato z [18].	6
Obr. 2.3:	Adresace na sběrnici I ² C [18].	6
Obr. 2.4:	Přenosový rámec sběrnice I ² C [18].	6
Obr. 2.5:	Příklad použití sběrnice I ² C – Teplotní senzor [15].	8
Obr. 2.6:	Zapojení sběrnice SMBus.	8
Obr. 2.7:	Vazby zařízení na sběrnici SMBus [12].	9
Obr. 2.8:	Časování přenosu sběrnice SMBus [13].	10
Obr. 2.9:	Časové limity na sběrnici SMBus [13].	11
Obr. 2.10:	(a) Send byte, (b) s kontrolním součtem [13].	12
Obr. 2.11:	(a) Receive byte, (b) s PEC [13].	12
Obr. 2.12:	(a) Write byte, (b) Write word [13],	12
Obr. 2.13:	(a) Read byte, (b) Read word [13].	13
Obr. 2.14:	Process call [13].	13
Obr. 2.15:	Časový průběh signálů na sběrnici USB1.1 [16].	17
Obr. 2.16:	Token paket.	18
Obr. 2.17:	Data paket.	18
Obr. 2.18:	Složení paketu Handshake.	18
Obr. 2.19:	Startovací paket.	18
Obr. 2.20:	Příklad zapojení portu USB a mikrokontroléru.	19
Obr. 3.1:	Přepět'ová ochrana – Zenerova dioda.	22
Obr. 3.2:	Ochrana proti nadproudu – vratná pojistka (polyswitch).	23
Obr. 3.3:	Obvod napájení.	23
Obr. 4.1:	Vývojový diagram programu mikrokontroléru.	29
Obr. 4.2:	Vývojový diagram datové části aplikace.	31
Obr. 4.3:	Vývojový diagram grafického rozhraní aplikace.	33
Obr. 5.1:	Provoz na datové lince SMB_DAT.	34
Obr. 5.2:	SMBus komunikace.	35
Obr. 5.3:	Periodická indikace „usbPoll()“	35
Obr. 5.4:	Provoz na sběrnici USB.	36

Obr. 5.5: Konektor akumulátoru – čelní pohled.....	38
--	----

SEZNAM TABULEK

Tab. 2.1:	Vlastnosti ATmega168 [2].....	4
Tab. 2.2:	Periferie ATmega168 [2].	5
Tab. 2.3:	Časování signálů sběrnice SMBus [13]	10
Tab. 2.4:	Vyhrazené adresy [13].	14
Tab. 2.5:	Stejnoseměrné rozdíly sběrnic SMBus a I ² C [13].	15
Tab. 2.6:	Třídy standardu USB [16].....	16
Tab. 5.1:	Testované akumulátory	37
Tab. 5.2:	Parametry testovaného akumulátoru	37
Tab. 5.3:	Zapojení konektoru na Smart Battery	38

ÚVOD

V dnešní době, díky prudkému rozvoji techniky a technologií, spatřují světlo světa rozličné neuvěřitelné objevy v různých oblastech vědy. Trendem elektrotechnologií je maximálně integrovat veškerý hardware do minimálních rozměrů v podobě miniaturních čipů. Není tedy divu, že se tyto malé čipy objevují dnes již snad v každém elektronickém zařízení od toustovačů přes moderní svářečky až k chytrým telefonům.

Svoje velké uplatnění mají i čipy určené k měření baterie, predikci zbylého náboje a uchování záznamů o baterii. Tyto čipy danou baterii neustále proměřují a na základě výsledků a znalosti stáří vyhodnocují její stav. Dále pak předpovídají zbylý náboj v baterii, měří její teplotu při nabíjení a uchovávají záznamy výrobce po celou dobu její životnosti. Napájené zařízení, využívající energii z baterie, musí s tímto měřicím čipem přes určité rozhraní komunikovat a získávat tak informace o vybíjení, nabíjení, zbývající energii, atd. Přenos dat probíhá po datové sběrnici. Typickým případem z praxe je přenosný počítač (notebook) a napájecí akumulátor s vestavěným měřicím hardwarem.

Tato diplomová práce se zabývá studiem těchto měřicích čipů v inteligentních akumulátorech. Obsahem práce je návrh a konstrukce diagnostického zařízení, které je schopné navázat spojení s akumulátorem – respektive s čipem uvnitř akumulátoru. Zkonstruované zařízení zprostředkovává sběr dat z inteligentního akumulátoru – Smart Battery. Pro zpracování a interpretaci těchto informací je použit osobní počítač, který je schopen podat uživateli výsledky ve srozumitelné podobě. V textu jsou nastíněny možné varianty realizace. Dle stanovených kritérií je zvolen jeden návrh, ten je popsán a realizován.

Z důvodu rozmanitosti typů inteligentních akumulátorů na trhu je vhodné, aby byl nakonec proveden test zařízení na několika odlišných akumulátorech. S rozmanitostí typů akumulátorů jde ruku v ruce i nejistota zapojení jejich konektoru. V samém závěru práce je uveden i popis několika typů konektorů na různých akumulátorech.

1 STRUKTURA SYSTÉMU

1.1 Požadavky na systém

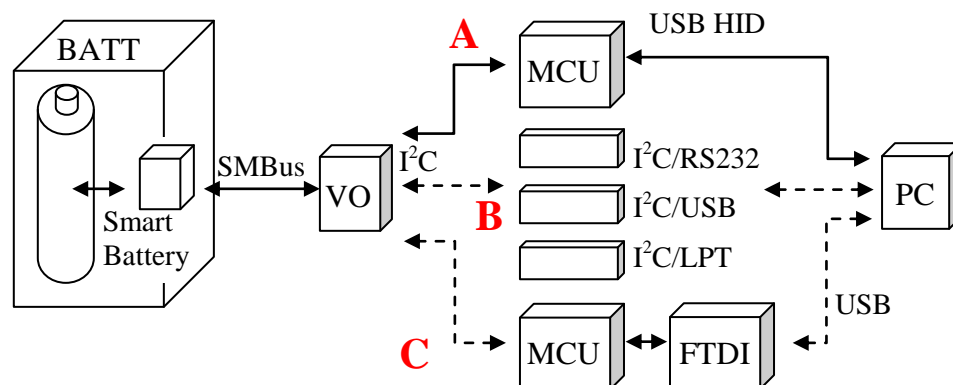
Je třeba si uvědomit základní požadavky na systém jako celek s ohledem na základní funkčnost, použitelnost, kompaktnost, univerzálnost a pohodlné ovládání navrhovaného systému.

- a) Základní funkčnost – je-li požadavkem, aby byla čtena a upravována data v chytrých akumulátorech „Smart Battery“, musí existovat zařízení, které s akumulátorem komunikuje. Prvním úkolem je spojení sběrnici SMBus s měřicím HW ve „Smart Battery“, druhým je zpracování a poskytnutí těchto dat uživateli v jakékoliv formě (zobrazení, přeposlání, atd.)
- b) Použitelnost – je nasnadě, aby systém fungoval jak u starších akumulátorů, tak u novějších. Existují různé standardy sběrnice SMBus, více v kap. 2.3. Tento problém musí být řešen vstupními obvody.
- c) Kompaktnost – kupuje-li zákazník/uživatel elektroniku, požaduje menší rozměry, nižší hmotnost a více funkcí. Tak je tomu i zde. Tento systém by měl být pokud možno jeden celek, bez dalších podpůrných zařízení.
- d) Pohodlné ovládání – téměř každý uživatel v dnešní době vlastní osobní počítač, nebo má alespoň k němu přístup. Ovládání a nastavování pomocí PC skýtá široké možnosti. Z hlediska pohodlí je více než vhodné, aby tento navrhovaný systém byl ovládán právě pomocí PC.
- e) Univerzálnost – ne každý PC obsahuje průmyslové porty (COM, LPT, GPIB, atd.). Na přenosných PC se vždy objevují porty USB, na které se připojuje většina periférií. Navržený systém by měl být připojitelný k PC pomocí USB portu ve třídě HID, viz kap. 2.4.

1.2 Teoretické možnosti realizace

Nástin celého systému ukazuje Obr. 1.1. Akumulátor obsahuje jednak řídicí HW (Smart Battery) a jednak samotný zdroj energie – tedy články. „Smart Battery“ komunikuje s okolím sběrnici SMBus, na kterou jsou připojeny i vstupní obvody (VO). Tento blok zajišťuje ochranu sběrnice SMBus od parazitních jevů, popř. převod napěťových úrovní signálu. Na konci linie je PC s obslužnou aplikací, která celý systém ovládá. Existuje více možností, jak převést data ze sběrnice SMBus na počítač. Na Obr.1.1 jsou nastíněny tři reálné, jednoduché a univerzální varianty.

Autor vychází z předpokladu, že sběrnice SMBus je se sběrnici I²C tak podobná, že je za určitých podmínek zaměnitelná. Sběrnice I²C je mnohem rozšířenější a leckteré mikrokontroléry ji mají už z výroby hardwarově implementovanou.



Obr. 1.1: Možnosti realizace systému.

- A. Mikrokontrolér, který má hardwarovou implementaci sběrnice I²C, se připojuje na jedné straně ke vstupním obvodům. Na druhé straně je připojen k portu USB ve standardu HID. Ovládací software pro mikrokontrolér se chová jako USB host. Zařízení se pak jeví jako softwarový převodník sběrnic I²C/USB.
- B. Je použit jeden, nebo více HW převodníků, které jsou naznačeny na Obr. 1.1. Na první pohled se jedná se o nejjednodušší způsob řešení. Aby bylo dosaženo univerzálnosti, byl by použit převodník sběrnic I²C/USB. Toto však skýtá nový problém, a to ovladače. Je-li k PC připojeno zařízení portem USB, musí k němu existovat i SW ovladače a právě s těmi bývá v praxi problém. V praxi se setkáváme s faktem, že se stejné ovladače chovají pod různými operačními systémy jinak, v horších případech nefungují vůbec.
- C. Na sběrnici I²C je připojen mikrokontrolér. Ten přeposílá data po sériové lince na převodník. Obvod FTDI se chová jako konvertor sběrnic RS232/USB. Po připojení k PC zastává funkci USB hosta. Nevýhodou však jsou opět ovladače operačního systému. Tento způsob má navíc další nevýhodu, a to nutnost jednoho dalšího bloku navíc. Tímto se zvyšuje cena, složitost, velikost desky plošného spoje a pravděpodobnost výskytu chyby.

Nejvhodnějším řešením se tedy jeví **varianta A**. Za cenu zvětšení objemu kódu pro mikrokontrolér (kvůli složitějšímu SW) nastává výrazné zjednodušení konstrukce a zároveň univerzálnost zařízení. Tato varianta řešení byla doporučena zadavatelem projektu „Interface diagnostické sběrnice pro akumulátory“ jako nejvhodnější. Výběr jiné varianty řešení projektu může nastat pouze při odhalení fatálních nedostatků použitého softwaru, či při zjištění závažných příčin použití třídy HID (rychlost přenosu, licence, atd.).

2 POPIS DÍLČÍCH ČÁSTÍ

2.1 Řídicí jednotka – MCU

2.1.1 Požadavky

Řídicí obvod navzájem propojuje dva typy sběrnic a ve skutečnosti se tedy jeví jako převodník. Při výběru vhodného mikrokontroléru hrají tyto nároky velkou roli:

- a) Z důvodu nárůstu velikosti kódu programu (virtuální implementace hosta USB – V-USB [19]) musí být paměť minimálně 4 kB velká.
- b) Pro komunikaci s PC je třeba vytvořit softwarové USB rozhraní, které pracuje s minimální taktovací frekvencí MCU $f_{\text{MIN}} = 12 \text{ MHz}$.
- c) Z toho samého důvodu musí být v řídicím obvodu alespoň jedno externí přerušení [19], které spouští komunikaci na sběrnici USB.

Zadavatelem projektu byl dodán mikrokontrolér ATmega168 od firmy Atmel corporation [2]. Tento obvod s přehledem přesahuje definované požadavky.

2.1.2 Popis ATmega168

Tento obvod z rodiny megaAVR se řadí do skupiny 8bitových RISC mikrokontrolérů. Lze jej programovat (zapisovat do paměti programu, nebo do přídavné EEPROM paměti) paralelně, rozhraním JTAG, nebo rozhraním SPI. Jeho vlastnosti jsou uvedeny v Tab. 2.1. Dostupnost a cena na trhu je dobrá.

Tab. 2.1: Vlastnosti ATmega168 [2].

Maximální pracovní frekvence	20 MHz
Paměť programu (Flash)	16 kB
Paměť dat EEPROM	512 B
Paměť dat SRAM	1 kB
Napájení	2,7 – 5,5 V
Pouzdro	PDIP28

Periferie obvodu jsou uvedeny v Tab. 2.2. Ty periferie, které budou v zařízení použity, jsou zvýrazněny tučným písmem.

Tab. 2.2: Periferie ATmega168 [2].

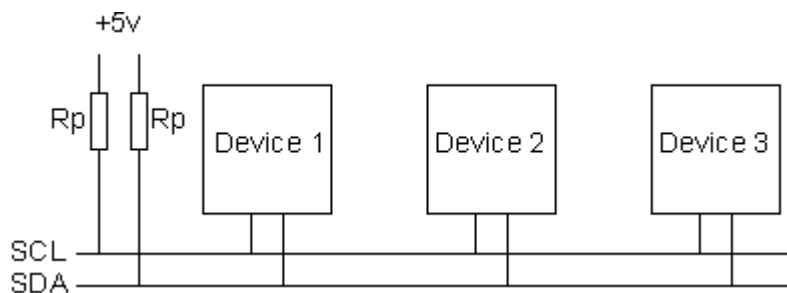
2 × 8bitový čítač/časovač
16bitový čítač/časovač
6 × PWM kanál
Sběrnice I ² C (TWI)
sběrnice USART
rozhraní SPI
Watchdog timer
2 × Externí přerušení
Analogový komparátor

2.2 Sběrnice I²C

2.2.1 Vlastnosti sběrnice I²C

Sběrnice I²C je dvou vodičová adresovatelná sběrnice, která umožňuje komunikaci mezi dvěma zařízeními s pevně danou adresou. Je založena na hierarchii „Master – Slave“. Tato sběrnice nabízí jednoduchost, zato však nižší přenosovou rychlost. Zapojení sběrnice je naznačeno na Obr. 2.1. Všechna zařízení jsou připojena vodiči:

- SDA – Synchronní data
- SCL – Hodinový signál
- GND – Společný nulový potenciál



Obr. 2.1: Zařízení na sběrnici I²C [18].

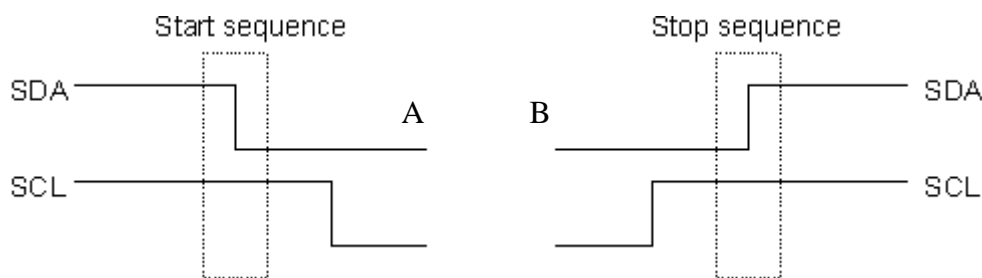
Spuštění komunikace má na starosti Master, ten vysílá adresu a požadavek. Ostatní zařízení (Slave) tedy pouze odpovídají. V jednoduchých implementacích sběrnice I²C je pevně dáno jedno hlavní zařízení (Master) a několik podřízených (Slave). Je však možné, aby existovalo více dočasných Masterů. Zařízení, které začíná vysílat, se od té chvíle stává Masterem a odesílá data. Po skončení se toto zařízení opět stává podřízeným, tedy Slave. V momentě kolize dvou Masterů přichází na řadu arbitráž, viz kap. 2.2.4.

2.2.2 Přenos dat

Výstupy všech zařízení na sběrnici musí být typu „otevřený kolektor“.

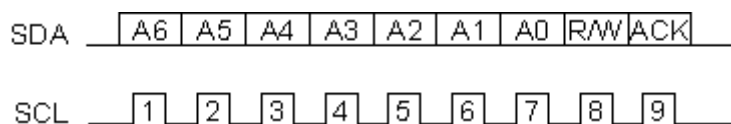
Při nastavení nízké logické úrovně se připojí SDA na GND, naopak při úrovni „H“ se linka SDA pouze odpojí. Vysokou úroveň zajistí zdvihací rezistory, připojené k napájecímu napětí, jak je patrné z Obr. 2.1. Signál SDA musí být v době čtení konstantní, jedná se o časový úsek, kdy je SCL v úrovni „H“.

Zařizováním, zahajujícím přenos dat, musí být vždy Master, ať již dočasný, nebo pevný. V době, kdy neprobíhá komunikace na sběrnici, jsou oba signály (SDA a SCL) v logické úrovni „H“. Prvním krokem Masteru je tzv. „Start podmínka“, viz Obr. 2.2(A). Přejít signálu SDA z vysoké logické úrovně na nízkou za podmínky úrovně „H“ na SCL je jedinečný, tedy s jakýmikoliv daty během přenosu nezaměnitelný.



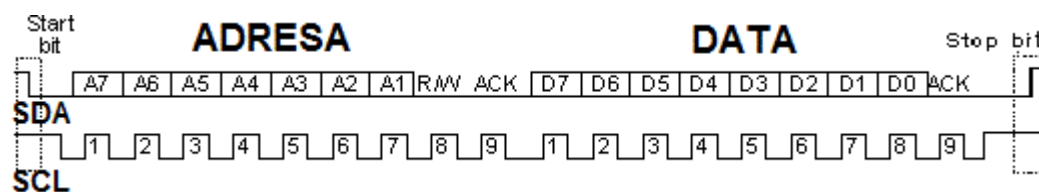
Obr. 2.2: A – Start podmínka, B – stop podmínka, převzato z [18].

Následuje adresace zařízení Slave. Tato data mají vždy délku 8 bitů, bit MSB je vyslán jako první. Ve většině případů má adresa délku 7 bitů a poslední bit určuje, zdali půjde o čtení ze Slave, nebo zápis (R/W), viz Obr. 2.3. Pak následuje potvrzení ze strany Slave (ACK) přidržetím linky SDA v úrovni „L“. Tímto je volané zařízení detekováno.



Obr. 2.3: Adresace na sběrnici I²C [18].

Po detekci zařízení a určení R/W je poslán Byte dat, MSB je opět první na řadě. Následuje detekce přijatého Bytu ACK. Posledním úkonem je uzavření přenosu podmínkou Stop. Ta je opět jedinečná, viz Obr. 2.2.B. Celý rámec přenosu je na Obr. 2.4.



Obr. 2.4: Přenosový rámec sběrnice I²C [18].

Je-li jedno zařízení pomalejší, může přidržetím SCL v úrovni „L“ prodloužit dobu čtení a zpracování. Tímto se aktuální rychlost přenosu přizpůsobí právě nejpomalejšímu zařízení, to vše se děje dynamicky za běhu.

2.2.3 Adresování zařízení

Na sběrnici je možné připojit až 128 různých zařízení s odlišnou adresou. Limitací je počet adresních bitů a maximální součet kapacit všech zařízení a vedení. Tento součet musí být menší než 400 pF [8]. Je tedy jasné, že i jediné zařízení, které je připojené dlouhým kabelem s vysokou kapacitou (blíží se 400pF), může silně degradovat sběrnici. Ve většině případů je adresa 7bitová, v určitých implementacích sběrnice I²C může být 8bitová, v některých dokonce 10bitová. Na sběrnici pak může být připojeno až 2¹⁰ zařízení, adresa je pak rozdělena na 2 + 8 bitů. Vše je opět limitováno maximální přípustnou kapacitou sběrnice.

Zařízení mají adresu buď pevně nastavenou, nebo zčásti nastavitelnou a zčásti pevnou. O přidělování adres se stará specializovaná komise, přidělování „Slave“ adres je zpoplatněno [8]. Ve standardu sběrnice I²C existuje několik rezervovaných adres pro speciální účely, jako např.:

- 000 0000 – tzv. broadcast/general call, na zprávu s touto adresou odpovídá každé zařízení na sběrnici
- 0000 001 – protokol C-Bus
- 111 11XX – hlavička 10bitové adresy
- 000 0011, 000 01XX – speciální účely

Adresa je posílána ihned za startovací podmínkou a je potvrzena (bit ACK), jak je vidět na Obr. 2.4. Pokud je adresa 10bitová, následuje po potvrzení zbylá část adresy (1 Byte).

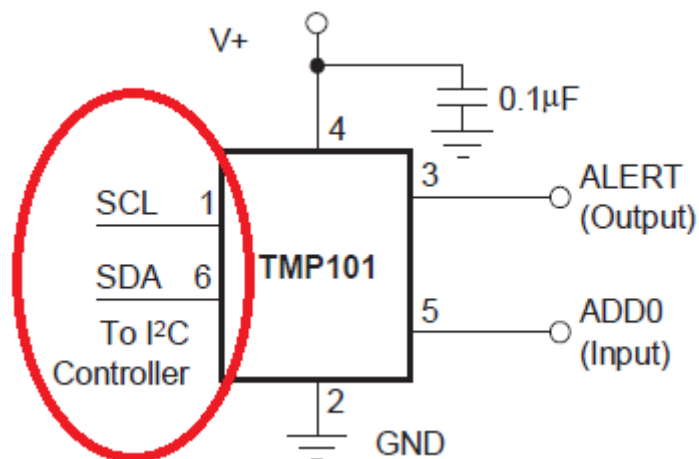
2.2.4 Arbitráž kolize

Zahajuje-li zařízení Master přenos dat, může nastat situace, že v tom samém momentě zahájí komunikaci i jiný dočasný Master na sběrnici. V tu chvíli probíhá korektní start podmínka. SDA linka plní funkci „wired-AND“, to znamená, že úroveň „H“ nastane pouze při vysoké úrovni současně všech zařízení na sběrnici. Při prvním rozdílném bitu obou je zjištěna kolize. Master, vysílající úroveň „H“, uvolní výstup, avšak na lince SDA detekuje ve stejnou chvíli úroveň „L“, kterou právě vysílá druhý Master. V tu chvíli se Dočasný master, který zjistil kolizi, odpojí.

Tato metoda řeší kolizi nedestruktivní metodou. Druhé zařízení pokračuje v komunikaci bez poškození dat.

2.2.5 Použití sběrnice I²C

Sběrnice I²C našla hojné uplatnění především v různých periferních obvodech, které se připojují k hlavnímu zařízení (kontroléru, PC, řídicímu systému). Datový tok směrem do periférií, nebo z nich, neklade vysoké nároky na rychlost přenosu. Ve většině případů jde o nastavování několika parametrů, měření teploty, měření otáček apod. Sběrnice I²C je tedy elegantním řešením. Rychlost přenosu bývá < 100 kb/s, v rychlém módu pak < 400 kb/s [6]. V praxi se toto rozhraní využívá např. v nízkorychlostních D/A a A/D převodnících, pro řízení OLED a LCD displejů, na čtení RTC obvodů a teplotních čidel, aj [8]. Na Obr. 2.5. je ukázka čipu pro řízení inteligentních reproduktorů.



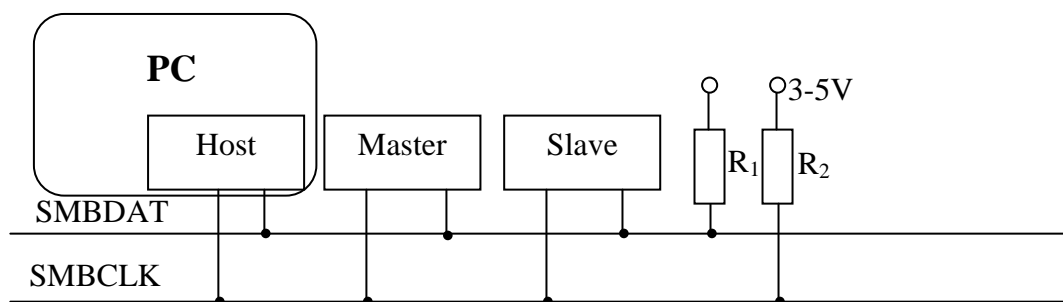
Obr. 2.5: Příklad použití sběrnice I²C – Teplotní senzor [15].

2.3 Sběrnice SMBus

2.3.1 Vlastnosti sběrnice SMBus

Sběrnice SMBus (popř. SMB) je sériová dvou vodičová sběrnice, která byla navržena a používána především pro ovládání zařízení typu Smart Battery a jiných napájecích systémů (Power Management). První verze byla definovaná v roce 1995 [5]. Její podstata vchází ze sběrnice I²C. Zařízení „Master“ odesílá data na ostatní podřízená zařízení po datové lince a zároveň generuje taktovací signál sběrnice. Zařízení „Slave“ se též může stát dočasným „Masterem“ a komunikaci řídit. Kolizi dvou Masterů na sběrnici řeší arbitráž, viz kap. 2.2.4.

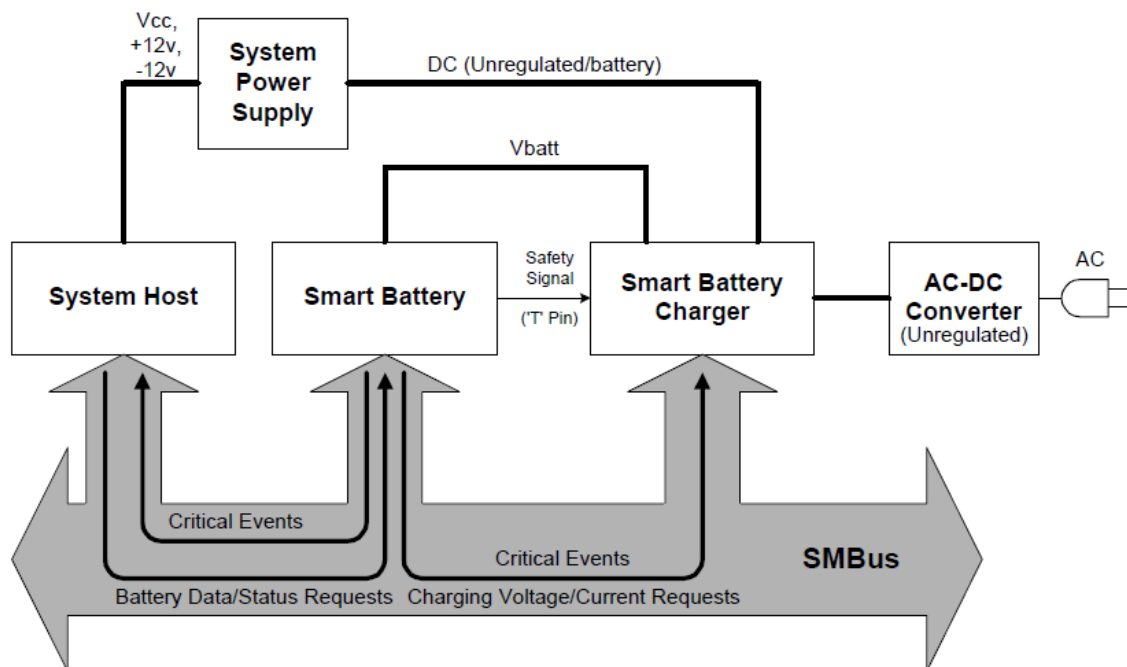
V systému navíc figuruje další zařízení „Host“ (hostitel). Tímto bývá zpravidla prvek, který komunikuje jak s napájecím zařízením Smart Battery, tak s napájeným zařízením (např. počítačem). Je to, zjednodušeně řečeno, datová spojka mezi objektem a jeho akumulátorem. Zapojení sběrnice je vidět na Obr. 2.6.



Obr. 2.6: Zapojení sběrnice SMBus.

Podmínka Start a podmínka Stop jsou totožné s protokolem sběrnice I²C. Data mohou změnit svoji úroveň pouze, je-li hodinový signál SMBCLK v úrovni „L“. V datovém rámci je nejprve posílána adresa zařízení „Slave“. Za ní následuje bit určující směr dat (R/W) a potvrzení (ACK) přesně tak, jak je tomu u sběrnice I²C, viz kap. 2.2. Komunikace po sběrnici SMBus probíhá pomocí protokolů, které udávají přesnou podobu rámců dat. Chce-li zařízení s implementovanou sběrnicí I²C po této

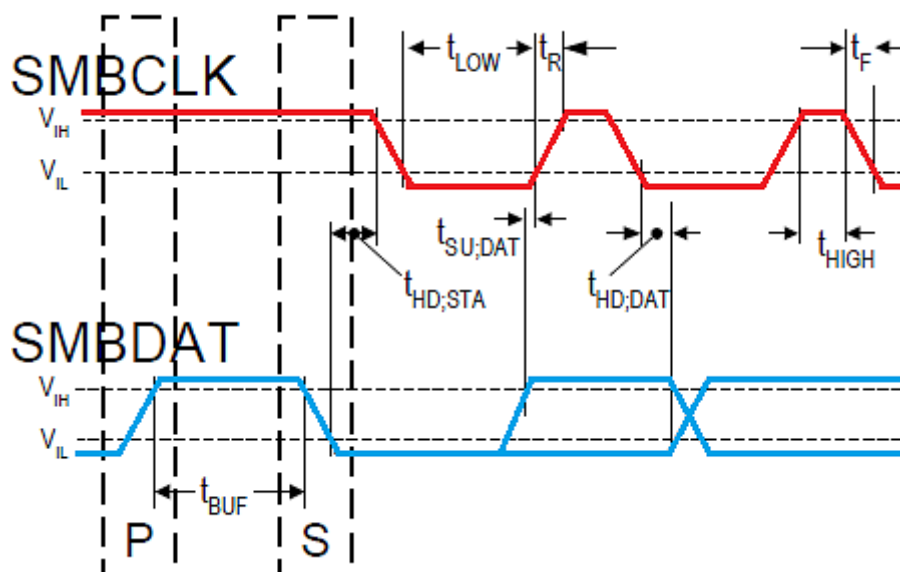
sběrnici komunikovat, musí tyto protokoly podporovat. Obr. 2.7. ukazuje vazby mezi zařízeními. Napájený systém, reprezentovaný „Hostem“, komunikuje se zařízením Smart Battery. Nabíječka (Charger) se stará o akumulátorové články dle pokynů a zároveň podává informace o aktuálním stavu článků (System Power Supply). AC-DC je výkonový prvek, který dodává energii v potřebné formě a potřebném množství.



Obr. 2.7: Vazby zařízení na sběrnici SMBus [12].

2.3.2 Časování přenosu dat

Frekvence hodinového signálu je omezena v rozsahu 10 – 100 kHz. Pro zaneprázdněný „Slave“ to tedy znamená ztížení podmínek. Příjímací zařízení nemůže protáhnout periodu SMBCLK libovolně dle vlastní potřeby (tím snížit rychlost), jak je tomu ve standardu sběrnice I^2C . Obr. 2.8. a korespondující Tab. 2.3. ukazují časování přenosu po sběrnici SMBus. Po skončení přenosu dat (P – podmínka Stop) následuje začátek nového přenosu (S – podmínka Start).

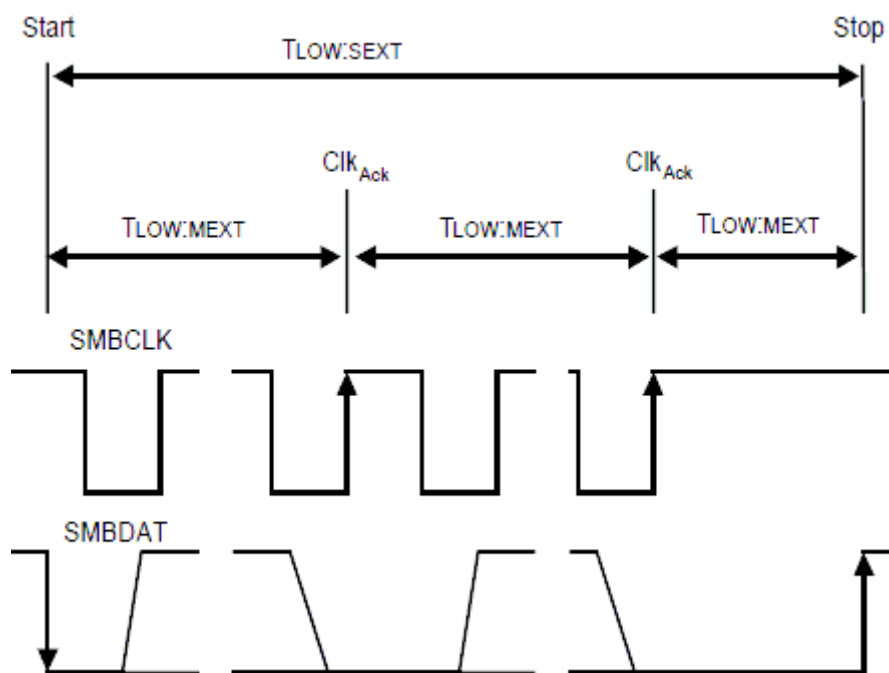


Obr. 2.8: Časování přenosu sběrnice SMBus [13].

Tab. 2.3: Časování signálů sběrnice SMBus [13]

Symbol	Parameter	Limits		Unit
		Min	Max	
F _{SMB}	SMBus Operating Frequency	10	100	kHz
T _{BUF}	Bus free time between Stop and Start Condition	4.7	-	μs
T _{HD:STA}	Hold time after (Repeated) Start Condition. After this period, the first clock is generated.	4.0	-	μs
T _{SU:STA}	Repeated Start Condition setup time	4.7	-	μs
T _{SU:STO}	Stop Condition setup time	4.0	-	μs
T _{HD:DAT}	Data hold time	300	-	ns
T _{SU:DAT}	Data setup time	250	-	ns
T _{TIMEOUT}	Detect clock low timeout	25	35	ms
T _{LOW}	Clock low period	4.7	-	μs
T _{HIGH}	Clock high period	4.0	50	μs
T _{LOW: SEXT}	Cumulative clock low extend time (slave device)	-	25	ms
T _{LOW: MEXT}	Cumulative clock low extend time (master device)	-	10	ms
T _F	Clock/Data Fall Time	-	300	ns
T _R	Clock/Data Rise Time	-	1000	ns
T _{POR}	Time in which a device must be operational after power-on reset	-	500	ms

Časové limity signálů při umělém snižování rychlosti za běhu jsou striktně dány. Maximální přípustné hodnoty a jejich význam ukazuje Obr. 2.9. v korespondenci s Tab. 2.3.



Obr. 2.9: Časové limity na sběrnici SMBus [13].

2.3.3 Protokoly sběrnice SMBus

Aby byla všechna zařízení na sběrnici schopna komunikace, musejí mít podporu tzv. protokolů SMBus již implementovanou. Protokoly jsou prototypy rámců, které definují velikost, směr a posloupnost přenášených dat.

Symboły na obrázcích:

S – Podmínka Start

Sr – Opakovaná start podmínka

P – Podmínka Stop

Wr – Zápis

Rd – Čtení

n (číslo nad rámcem) – počet bitů v segmentu

A – potvrzení (0 = potvrzeno, 1 = konec komunikace)

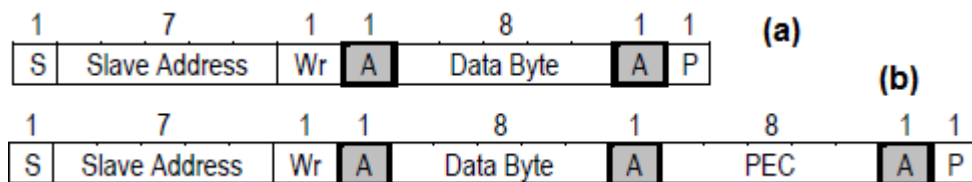
PEC – kontrolní součet (Packet error code)

□ – Přenos Master → Slave

■ – Přenos Slave → Master

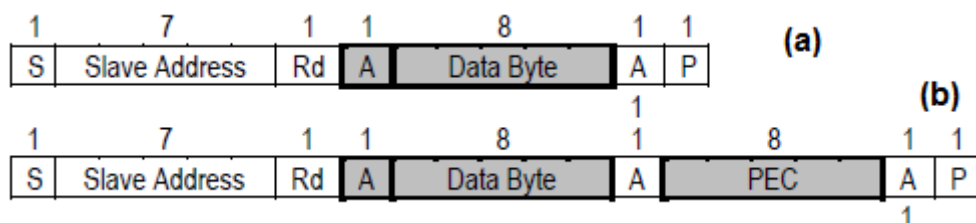
Specifikace sběrnice SMBus udává tyto typy protokolů:

- Send Byte – adresovaný „Slave“ obdrží jeden byte. Prvních 7 bitů lze například využít určení registru, poslední bit pak nastavuje on/off. Na Obr. 2.10 je zobrazen rámeček s kontrolním součtem PEC i bez něj.



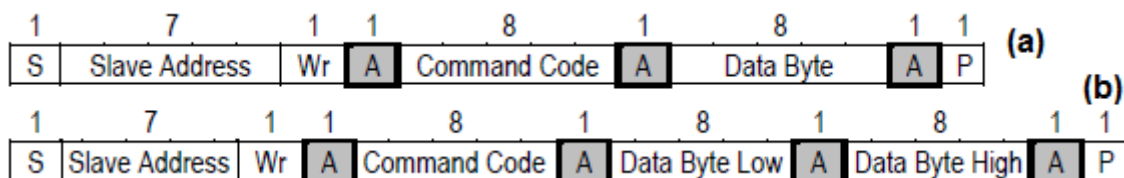
Obr. 2.10: (a) Send byte, (b) s kontrolním součtem [13].

- Receive byte – převrácený případ oproti Send byte. Zařízení „Slave“ poskytuje data pro „Master“ v podobě jednoho byte, viz Obr. 2.11.



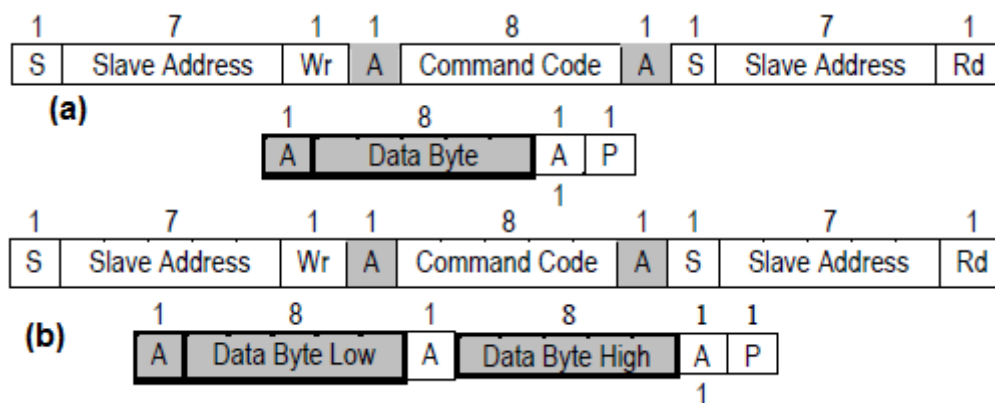
Obr. 2.11: (a) Receive byte, (b) s PEC [13].

- Write byte/word – prvním bytem je příkazový kód, ostatní 1 – 2 byty jsou data, která mají být zapsána. Vše je ukázáno na Obr. 2.12. Do obou variant je možné přidat kontrolní PEC kód.



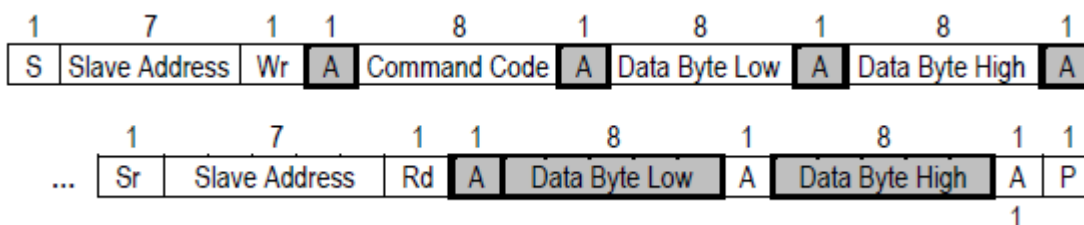
Obr. 2.12: (a) Write byte, (b) Write word [13],

- Read byte/word – čtení dat ze „Slave“ je komplikovanější než zápis. Jako první je poslán příkazový kód směrem k zařízení „Slave“, pak následuje znovuspuštění komunikace podmínkou Start, adresace a načtení dat, viz Obr. 2.13. Kontrolní součet (PEC) na konci řetězce je volitelný.



Obr. 2.13: (a) Read byte, (b) Read word [13].

- Process call – „Master“ odešle několik příkazů za sebou a po obnovení Start podmínky očekává data od zařízení „Slave“, viz Obr. 2.14. Kontrolní PEC je volitelný.



Obr. 2.14: Process call [13].

- Další protokoly: Quick command, Block Read/Write, Block Write – Block Read Process call, Block Write – Block Read Process call, Host notify protocol

2.3.4 Adresace zařízení

Pro identifikaci zařízení na sběrnici jsou použity jedinečné adresy nazývané „Slave address“. Rezervované adresy jsou vyhrazeny pro speciální využití a nelze je použít pro identifikaci zařízení. Výčet adres ukazuje Tab. 2.4. V rámci zařízení „Interface diagnostické sběrnice pro akumulátory“ bude využito vyhrazených adres „SMB Host“, „Device default address“, „Smart Battery“, případně „General Call“ a „Start byte“.

Tab. 2.4: Vyhrazené adresy [13].

Slave address (Bits 7-1)	R/W (Bit 0)	Comment
0000 000	0	General call address
0000 000	1	START byte
0000 001	X	CBUS address
0000 010	X	Address reserved for different bus format
0000 011	X	Reserved for future use
0000 1XX	X	Reserved for future use
0101 000	X	Reserved for ACCESS.bus host
0110 111	X	Reserved for ACCESS.bus default address
1111 0XX	X	10-bit slave addressing
1111 1XX	X	Reserved for future use
0001 000	X	SMBus Host
0001 001	X	Smart Battery Charger
0001 010	X	Smart Battery System Manager
0001 011	X	Smart Battery
0001 100	X	SMBus Alert Response Address
1100 001	X	SMBus Device Default Address

2.3.5 Packet Error Code

Posloupnost vyslaných/přijatých dat lze zabezpečit kontrolním součtem PEC. Jedná se o redundantní data, která nenesou žádnou informaci. Mají za úkol odhalovat chyby vzniklé přenosem. Tento kód vzniklé chyby neopraví, pouze je detekuje.

Dle specifikace je velikost PEC kódu 8 bitů (1 byte), které se nachází na konci celého přeneseného rámce. Použití je sice volitelné, avšak především u protokolu Block write – Block read Process call (velký objem přenesených dat) je použití PEC vysoce doporučeno [13].

Kód PEC je vypočítán kódováním CRC–8–ATM HEC přes všechny bity včetně adresy a R/W bitu.

Generující polynom: $x^8+x^2+x^1+1$ [14].

2.3.6 Rozdíly mezi standardy sběrnic SMBus a I²C

- Napěťové a proudové rozdíly: sběrnice I²C především může být napájena pouze napětím 5V. Dále má jiné, pevně dané rozhodovací úrovně. Napětí rozhodovacích úrovní je 1,5 a 3 V. Oproti tomu sběrnice SMBus má napětí rozhodovacích úrovní jiné, a to 0,8 V a 2,1 V. Díky tomu může být napájecí napětí sběrnice zvoleno v rozsahu 3 – 5 V, viz Tab. 2.5.
Proud I_{OL} teče vodiči SMBDAT/CLK v době, kdy je na lince úroveň „L“ což odpovídá napětí 0,4 V. Pro nízkopříkonové systémy je třeba maximální hranici I_{OL} dodržet kvůli vlastnímu vybíjení článků Smart Battery [13].

V systému „Interface diagnostické sběrnice pro akumulátory“ na tento fakt není třeba brát ohled.

Dalším kritériem je proud zdvihacími rezistory I_{PULLUP} a únikový proud I_{LEAK} , který teče z/do neaktivního zařízení (zapnuté nebo vypnuté zařízení na sběrnici). Ani tyto parametry nyní nehrají žádnou roli.

Tab. 2.5: Stejnoseměrné rozdíly sběrnic SMBus a I²C [13].

Parametr	I ² C		SMBus		Jednotka
	Min	Max	Min	Max	
V_L	-0,5	1,5	-	0,8	V
V_H	3	$V_{DD} + 0,5$	2,1	5,5	V
$I_{OL} (V_{OL}=0,4V)^*$	-	3000	-	350	μA
I_{PULLUP}^*	-	-	100	350	μA
I_{LEAK}^*	-10	10	-5	5	μA

* Kriterium lze pro toto zařízení zanedbat.

- Časové odlišnosti: sběrnice SMBus určuje minimální a maximální frekvenci hodinového signálu SMBCLK, a to 10 – 100 kHz. Důvodem je presence zařízení na sběrnici. Není-li připojené zařízení schopno tuto frekvenci dodržet, znamená to, že má buď poruchu, nebo jiný problém. Komunikace nebyla dokončena. Dále je specifikován čas ($t_{HD:DAT} = 300$ ns) mezi sestupnou hranou hodinového signálu SMBCLK a změnou dat na SMBDAT. Protokol sběrnice I²C tento čas nedefinuje, je tedy roven 0 sekund. SMBus definuje maximální přípustný čas linky SMBCLK ve stavu „L“ na 35 ms, kdežto sběrnice I²C povoluje nekonečný čas. Dále je specifikací SMBus dán maximální součet časů, o který může „Slave“ protáhnout hodinový signál od podmínky Start do Stop ($t_{LOW:SEXT} = 25$ ms). Stejně tak je dán i limit součtu časových prodlev přidávaných „Masterem“ do hodinového signálu ($t_{LOW:SEXT} = 10$ ms). Jako poslední jsou specifikovány časy náběžných a sestupných hran [13]. Tyto podmínky protokol sběrnice I²C nedefinuje vůbec.
- Použití potvrzení ACK: když zařízení „Slave“ na sběrnici I²C nemůže data přijímat, nepotvrdí žádost „Masteru“. To může nastat například z důvodu zpracovávání jiné úlohy. Nebo může „Slave“ nepotvrdit jakýkoliv jiný následující Byte a tím přenos zastavit. Sběrnice SMBus stanovuje, aby byl požadavek potvrzen vždy, a to z důvodu detekce zařízení na sběrnici. Kromě této indikace zařízení je ACK navíc indikací neplatných dat či neplatného příkazu. Chyba může nastat i v posledním bytu, proto musí zařízení na sběrnici SMBus potvrzovat každý byte včetně PEC kódu [13].
- Posledním rozdílem jsou protokoly sběrnice SMBus, viz kap.2.3.3. Má-li být zařízení podporující I²C připojeno ke sběrnici SMBus, musí být protokoly dodrženy. Zvolený mikrokontrolér ATmega168 sice programátorovi nabízí nastavení parametrů sběrnice I²C, ne však v takové míře, aby mohly být sestaveny protokoly SMBus. **To je hlavní důvod, proč nelze využít periférii I²C implementovanou v mikrokontroléru ATmega168.** Protokoly tedy musí být sestaveny softwarově – bit po bitu.

2.4 Sběrnice USB

Sběrnice USB je univerzální sériová sběrnice, jejíž port je možno najít dnes již na každém stolním či přenosném počítači. Je standardizováno několik verzí tak, jak vývoj této sběrnice postupoval. Projekt „Interface diagnostické sběrnice pro akumulátory“ se do detailu zabývat rozbořem sběrnice USB nebude. Budou pouze nastíněny možnosti sběrnice USB a informace o USB, které bude tento projekt používat.

2.4.1 Verze sběrnice USB a jejich rozdíly

První verze sběrnice USB byla standardizována v roce 1996. Rozdílem mezi verzemi je především rychlost přenosu dat, maximální délka připojovacího kabelu a případné další rozšířené schopnosti.

- Ver.1.0 – Nabízí rychlost přenosu Low-Speed (1,5 Mb/s) a Full-Speed (12Mb/s). Není povoleno použít delší prodlužovací kabely.
- Ver.1.1 – Odstraněny chyby z předchozí verze.
- Ver.2.0 – Rychlost přenosu (High-Speed) dosahuje až 480 Mb/s. Je použito jiné kódování, umožňuje vzájemnou komunikaci dvou zařízení bez zásahu hostitele (PC), nabíjení baterií s proudem až 1,5 A, atd [16].
- Ver.3.0 – Umožňuje rychlost přenosu až 5 Gb/s (SuperSpeed), komunikace je na rozdíl od předchozích verzí plně duplexní, je použita expanze 8 bitů na 10 bitů [16].

2.4.2 Třídy USB zařízení

Aby hostitel (počítač) rozeznal funkci zařízení a přistupoval k němu správně, jsou definovány třídy zařízení. Každá třída má identifikační číslo a zahrnuje určitý okruh zařízení s podobnou funkcí. Některé třídy mají podporu v operačních systémech, jiné třídy pracují pouze s individuálním ovladačem [16]. Některé z tříd jsou ukázány v Tab. 2.6.

Tab. 2.6: Třídy standardu USB [16].

Číslo (hexadecimálně)	Třída	Příklad zařízení
01	Audio	Mikrofon, sluchátka
03	HID	Klávesnice, myš, ...
06	Image	Webkamera, scanner
07	Printer	Tiskárna
08	Mass storage	Flash disk, přehrávač
09	USB hub	Rozbočovač
0D	Content security	Čtečka otisků
E0	Wireless controller	Adaptér Bluetooth
FF	Vendor-specific	Speciální zařízení

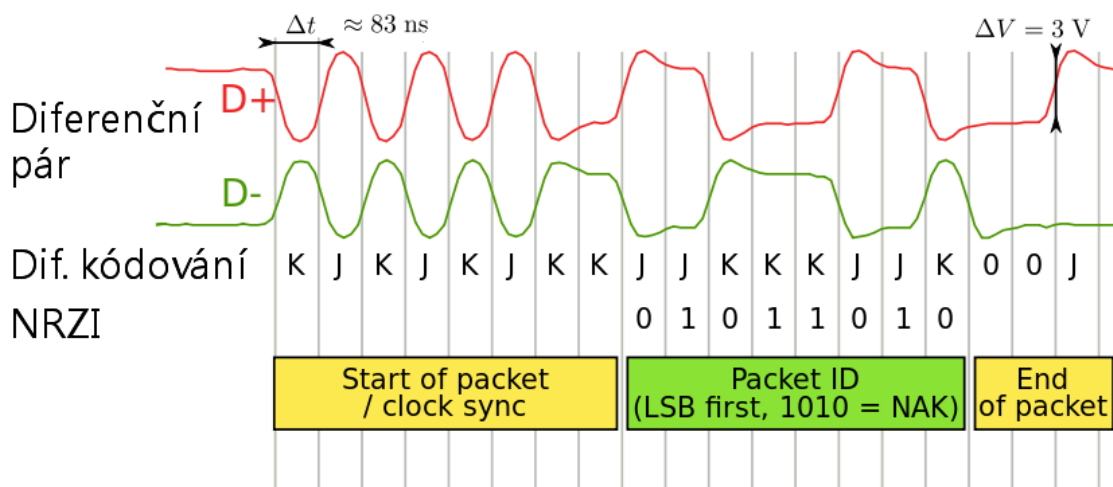
Zařízení projektu „Interface diagnostické sběrnice pro akumulátory“ bude pracovat ve třídě HID (Human Interface Device), která má podporu u převážné většiny operačních systémů. Pro komunikaci s PC není nutná dodatečná instalace ovladačů.

2.4.3 Signály na sběrnici USB

Spojení dvou zařízení je realizováno kabelem o čtyřech vodičích se stíněním. Dva vodiče jsou signálové, s charakteristickou impedancí $90\ \Omega$ a dva jsou napájecí. Přenos dat na sběrnici USB zajišťuje diferenční pár vodičů D+ a D-. Přenos se tak stává odolnější vůči rušení. Silné úzkopásmové rušení se superponuje na oba dva vodiče stejně a objeví se na nich vyšší napětí vůči zemnímu potenciálu. Rozdíl (diference) mezi vodiči D však zůstává přibližně konstantní – bez rušení. Tohoto využívají například protokoly RS485 a RS422, které jsou velmi vhodné pro vedení signálu v průmyslových (rušených) zónách.

Na Obr. 2.15 je naznačen časový průběh signálů na sběrnici USB1.1. Frekvence je 12 MHz a napětí diferenčních signálů ve stavu „H“ je 3,3 V. Úroveň „H“ ve starších verzích USB1.0 a 1.1 nabývá hodnot napětí 2,8 – 3,6 voltů. Oproti tomu vysoká úroveň módu High-Speed (USB2.0 a 3.0) je detekována v intervalu 0,36 až 0,44 voltů, což je markantní rozdíl. Zařízení s vyššími verzemi USB jsou navržena tak, aby tento fakt akceptovala a přizpůsobila se i starším zařízením [16].

Datový kabel k zařízení ve verzi USB1.0 a USB1.1 není zakončen, jedná se o otevřený konec – nekonečná impedance. Kabel k zařízení verze USB2.0 a vyšší musí být zakončen daným odporem za účelem eliminace odrazů na vedení. Kvůli nízkým napěťovým úrovním (verze USB2.0) by odraz vlny od nepřizpůsobeného konce vedení degradoval data přespříliš. Buď jsou oba diferenční vodiče uzemněny obecným odporem $45\ \Omega$ každý zvlášť, nebo mají navzájem mezi sebou odpor $90\ \Omega$ [16]. Přizpůsobení vedení lze realizovat např. rezistory.



Obr. 2.15: Časový průběh signálů na sběrnici USB1.1 [16].

Data jsou dekodována diferenčně na formu NRZI. To znamená, že změna symbolu odpovídá úrovni „L“ a zachování symbolu do příští periody znamená úroveň „H“.

2.4.4 Komunikace na sběrnici USB

Protokol USB používá pro přenos dat pakety. Jednotlivé pakety se skládají z dílčích polí [17]. Existují tato různá pole:

- Sync (synchronizace) – objevuje se na začátku každého paketu. Slouží k synchronizaci přijímače. Délka je 8/32 bitů (nižší/vyšší verze).

- PID – identifikační číslo paketu. Toto číslo určuje, o jaký typ paketu se jedná. Každý typ má vlastní PID číslo, např.: 0011 – data, 0010 – potvrzení dat atd.
- ADDR – adresa zařízení, pro které je paket určen.
- ENDP – koncové znaky dat. Toto pole má velikost 4 bity.
- CRC – cyklický redundantní součet provedený nad paketem.
- EOP – ukončení paketu, viz Obr. 2.15 (End of packet).

Z polí popsaných výše jsou sestavovány čtyři možné typy paketů pro různé účely [17]:

- a) Token paket – indikuje, jaký typ přenosu bude následovat, viz Obr. 2.16.

Sync	PID	ADDR	ENDP	CRC	EOP
------	-----	------	------	-----	-----

Obr. 2.16: Token paket.

- b) Data paket – přenos dat. Strukturu paketu naznačuje Obr. 2.17.

Sync	PID	Data	CRC	EOP
------	-----	------	-----	-----

Obr. 2.17: Data paket.

- c) Handshake paket – slouží k potvrzení přijatých dat. , viz Obr. 2.18.

Sync	PID	EOP
------	-----	-----

Obr. 2.18: Složení paketu Handshake.

- d) Start of frame packet – indikuje start nového rámce. Obsahuje 11b číslo rámce. Všechna pole tohoto paketu ukazuje Obr. 2.19.

Sync	PID	Frame number	CRC	EOP
------	-----	--------------	-----	-----

Obr. 2.19: Startovací paket.

Zařízení stále posílá startovací pakety. Jakmile vyšší vrstvy protokolu požádají o vyslání/přijem dat, je při nejbližším startu rámce (Start of frame paket) otevřena komunikace mezi hostitelem a daným zařízením. Následuje nastavovací paket Token, dále pak Data paket a poslední je potvrzovací Handshake paket [17].

O správu polí paketů se starají vyšší vrstvy protokolu. Tím jsou např. výpočty CRC, určení adresy, číslo rámce, atd. V případě hostitele (PC) jsou tyto informace zpracovávány ovladačem a řídicí aplikací. Pro zpracování dat v připojeném zařízení bude v projektu „Interface diagnostické sběrnice pro akumulátory“ použit volně přístupný software V-USB, viz kap.2.5.1.

2.5 Ovládací software

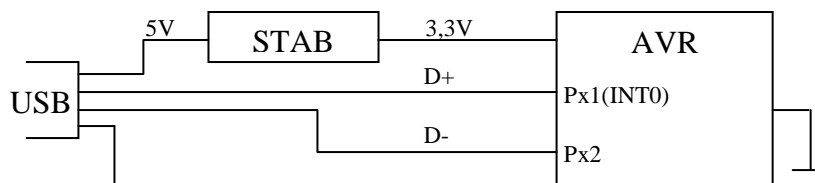
2.5.1 V-USB – virtuální USB port na mikrokontroléru

Tento volně šiřitelný software je dílem firmy Objective Development Software GmbH, která jej dává (nepatrně omezený) pod licencí GNU k dispozici. Komerční licence je plnohodnotná, bez omezení, však pro zařízení „Interface diagnostické sběrnice

pro akumulátory“ naprosto zbytečná. Tento software je možné získat z webových stránek firmy [19].

Prakticky jde o kód pro implementaci virtuálního portu USB na mikrokontroléry AVR, který je napsán v programovacím jazyce C. Veškerá zpracování signálů všech vrstev se dějí softwarově. Virtuální port je v souladu s verzí sběrnice USB1.1 a pracuje v režimu Low-Speed s maximální rychlostí přenosu 1,5 Mb/s. Maximální velikost přeneseného bloku dat jednorázově je 256 B. Velikost kódu pro paměť programu mikrokontroléru AVR je přibližně 2 kB a taktovací frekvence musí být buď 12, 15, 16, nebo 20 MHz [19].

Při návrhu plošného spoje zařízení musí být splněny určité požadavky týkající se napájení, napěťových úrovní a zapojení pinů na pouzdře kontroléru. Software V-USB využívá k indikaci přenosu dat externí přerušení mikrokontroléru. Proto vzniká podmínka, aby byl signál D+ portu USB připojen právě k pinu, který odpovídá externímu přerušení kontroléru. Dále je nutné zabezpečit napěťové úrovně datových signálů na maximální limitní hodnotu 3,3 V [19]. To lze realizovat buď omezovačem (např. zenerova dioda), nebo snížením napájecího napětí celého obvodu, viz Obr. 2.20.



Obr. 2.20: Příklad zapojení portu USB a mikrokontroléru.

Software V-USB se skládá z několika souborů, přičemž hlavní soubor ovladače využívá funkce z ostatních souborů. V konfiguračním souboru (usbconfig.h) je uloženo číslo výrobce, název zařízení, proud portem (odběr), třídu USB, zapojení pinů, atd. Tato data definuje výrobce zařízení před vlastním naprogramováním AVR kontroléru.

2.5.2 MS Visual Studio a jeho aplikace

Software Microsoft Visual Studio je vývojové prostředí, nabízející programátorovi široké možnosti, ať již v oblasti síťových aplikací, nebo desktopových (lokálních) aplikací. Co se týká nabídky programovacích jazyků, je možné si vybrat z jazyků Visual Basic, C++ a C#.

Program Visual Studio umožňuje vývoj aplikací na platformě .NET, která je velice rozšířená jak v PC, tak v inteligentních mobilních telefonech a PDA zařízeních [1]. Pomocí .NET bude přístupováno k portu USB, čtena data, zpracována a zobrazena.

Softwarová platforma .NET je zastřešující název pro soubor mnoha softwarových prostředků, který se stále rozšiřuje. Základní komponentou je MS .NET Framework, což je prostředí pro běh aplikací a využívání knihoven. První verze (1.0) byla firmou Microsoft Corporation představena v roce 2002. Nejnovější verze má označení 4.0 [1].

V projektu „Interface diagnostické sběrnice pro akumulátory“ bude využita verze Visual Studio 2008 [10], a to s licencí freeware. V této volně šiřitelné licenci je několik málo omezení, která mohou být překážkou v rozsáhlejších projektech. Omezení však nijak nevede vývoji nenáročných aplikací, která bude číst, zapisovat a zpracovávat data z portu USB.

Visual Studio 2008 pracuje s nejrůznějšími dynamickými (.dll) knihovnami, jako jsou knihovny pro grafiku webových stránek, knihovny pro servery, databáze, ovládání periférií, atd. Projekt „Interface diagnostické sběrnice pro akumulátory“ bude využívat knihovnu funkcí pro ovládání portů USB.

3 NÁVRH ZAŘÍZENÍ

Jak již bylo zmíněno výše (viz kap.1.2), zařízení se bude skládat z mikrokontroléru, obvodu pro napájení a vstupně-výstupních obvodů. Schéma se nachází v příloze A.1.

3.1 Celkové zapojení

Centrálním prvkem celého zařízení je mikrokontrolér ATMega168, který byl doporučen a po sléze dodán zadavatelem diplomové práce „Interface diagnostické sběrnice pro akumulátory“. Ke spojení zařízení s PC, tedy ke komunikaci na sběrnici USB, je do schématu přidána přípojka v podobě USB konektoru. Do něj se bude zapojovat kabel přímo z PC. Spojení zařízení s akumulátorem, tedy připojení na sběrnici SMBus, může být provedeno libovolným konektorem o třech kontaktech.

Obvody pro tvarování signálu, které se nachází mezi konektorem USB a mikrokontrolérem jsou určeny manuálem V-USB [19]. Jedná se o Zenerovy diody, které spolu s rezistory omezují maximální napětí logické úrovně „H“. Celkové schéma je uvedeno v příloze A.1.

Úpravu, ochranu a filtraci napájecího napětí zajišťuje napájecí obvod, který je podrobně popsán v kap. 3.3.

Komunikace na sběrnici USB požaduje taktovací frekvenci 12 MHz, což je zajištěno krystalem o téže frekvenci. Zapojení se dvěma kondenzátory o velikosti 22 pF je dáno manuálem mikrokontroléru [2]. Sběrnice USB je připojena na dva vstupně-výstupní piny mikrokontroléru. Jsou to konkrétně piny 1 a 2 portu D, přičemž signál D+ musí být připojen na pin s funkcí externího přerušení (INT) [19].

Nastane-li problém v komunikaci na sběrnici USB (mezi PC a zařízením), je nejlepším řešením toto spojení obnovit – restartovat. Aby nebylo nutné vysunout a zasunout konektor (což znamená odpojení napájení), je do obvodu přidáno resetovací tlačítko. Na resetovacím pinu mikrokontroléru ATMega168 [2] se při stisku tlačítka objeví logická „L“, tím je aktivován reset obvodu, a to bez nutnosti odpojení napájení.

Do zapojení sběrnice SMBus jsou přidány ochranné Zenerovy diody, zdvihací a sériové rezistory. Zdvihací rezistory udržují uvolněné linky SMBus v logické úrovni „H“. Sériové rezistory se při standardní komunikaci chovají jako propojka (mají velice nízký odpor). Při přepětí, např. ESD výboji [20], se Zenerova dioda zachová jako zkrat, proud rezistorem se rapidně zvýší a veškeré napětí výboje ESD se objeví právě na tomto sériovém rezistoru. Mikrokontrolér je tím pádem ochráněn. Sběrnice SMBus je realizována softwarově z důvodů popsaných v kap. 2.6.3. Signály SMBCLK a SMBDAT mohou tedy být realizovány kterýmkoliv vstupně-výstupním pinem na pouzdře mikrokontroléru. Změna zapojení pinů je otázkou řídicího programu. V tomto případě byly zvoleny piny č. 6 a 7 na portu D, návrh desky si to tak vyžadoval.

Posledními diskrétními prvky ve schématu jsou LED diody. První LED (zelená) indikuje přítomnost napětí – zapojení do portu USB, druhá LED (červená) indikuje provoz dat na zařízení.

Jelikož je projekt „Interface diagnostické sběrnice pro akumulátory“ unikátní záležitost, nepředpokládá se, že bude toto zařízení vyráběno továrně. Vzniká tedy

potřeba ladění softwaru přímo na DPS. K zápisu programu do mikrokontroléru je využito sériové rozhraní SPI. Do schématu je tedy přidán konektor, kterým se bude programátor připojovat k mikrokontroléru a ladit jej. Konektor je označen „PROG_C“. Seznam všech součástek je uveden v příloze A.4.

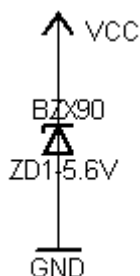
3.2 Ochranné obvody

Pro zvýšení odolnosti zařízení vůči okolním nežádoucím elektromagnetickým vlivům je třeba do zapojení přidat ochranné prvky. Ty zvyšují spolehlivost a životnost zařízení. Lze je dle účinku rozdělit na přepětové a nadproudové.

3.2.1 Ochrana proti ESD

Tato ochrana se řadí mezi ochrany přepětové. Výboj ESD vzniká přiblížením dvou nabitých těles, popř. nabitého a uzemněného tělesa. Takovým tělesem může být například lidské tělo. Při dotyku lidských rukou s přístrojem může dojít k výboji a následné destrukci vnitřních obvodů. Poruchy, způsobené tímto jevem, nejsou viditelné a jejich odstranění je velmi obtížné. Dle [20] při výboji vzniká napětí až několik desítek kV.

V obvodu jsou zařazeny nelineární prvky. Jsou to reverzně polarizované Zenerovy diody, viz Obr. 3.1. Při vyšším napětí se Zenerova dioda chová jako zkrat. Při výboji se proud diodou zvýší, ale napětí zůstává na prahové hodnotě Zenerovy diody. Tím dioda ochrání ostatní prvky v obvodu.



Obr. 3.1: Přepětová ochrana – Zenerova dioda.

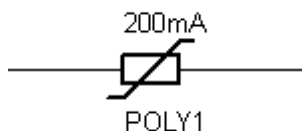
Tyto prvky se nachází všude tam, kde dojde ke styku mezi zařízením a lidskou rukou. Nejčastěji jsou to konektory a porty. V tomto zařízení jsou Zenerovy diody zapojeny na vstupně-výstupní piny sběrnice SMBus, na datové vodiče portu USB a napájecí vodiče. Na datových vodičích sběrnice USB tyto Zenerovy diody plní zároveň funkci omezovače napětí. Logické úrovně „H“ a „L“ jsou +3,6 V a 0 V. Toto je standardizované napětí logických úrovní pro mód LowSpeed USB [16].

3.2.2 Ochrana proti nadproudu

Tato ochrana je jednou ze základních ochran každého elektrického stroje, přístroje či zařízení. Primární nadproudová ochrana bývá zařazena mezi zdroj elektrické energie a vlastní zařízení. Jedná-li se o velmi komplikované zařízení s dílčími částmi, je vhodnější použít rozšířenou nadproudovou ochranu, a to pro každou část či blok.

V zařízení „Interface diagnostické sběrnice pro akumulátory“ bude použita pouze

základní nadproudová ochrana. Zařízení je napájeno z portu USB, proto se tato ochrana nachází ihned za konektorem USB. Na trhu jsou k dostání rozličné variace proudových ochran. Mezi nejjednodušší a nejlevnější ochrany patří přístrojová (tavná) pojistka. Z hlediska velikosti a údržby byla jako základní proudová ochrana vybrána tzv. vratná pojistka – polyswitch, viz Obr.3.2.



Obr. 3.2: Ochrana proti nadproudu – vratná pojistka (polyswitch).

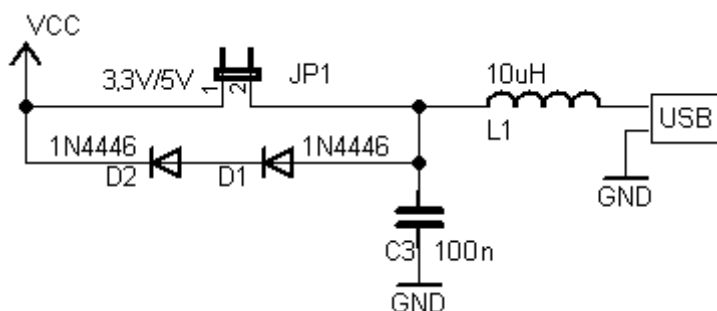
Polyswitch je ve své podstatě nelineární pozitivní termistor, který má při 20 °C téměř nulový odpor [21]. Jakmile protékající proud překročí jmenovitou hodnotu, začne se vratná pojistka prudce ohřívat a zvyšovat svůj odpor až na jednotky MΩ. Tím se pojistka stává rozpojenou. Po ochlazení se vratná pojistka navrátí do původního stavu.

S přihlédnutím na spotřebu mikrokontroléru [2] a přičtením rezervy, byla jmenovitá hodnota proudu vratné pojistky zvolena 200 mA.

3.3 Napájecí obvod

Z kap. 3.2.2 vyplývá, že spotřeba el. energie celého zařízení příliš vysoká nebude. Maximální odebíraný proud je 200 mA. Napájení obvodu externím stabilizovaným zdrojem by bylo nevýhodné, a to z důvodu jak ceny, tak i praktičnosti. Přibylo by navíc další zařízení, které je nutno připojit. Je tedy nasnadě využít napájení portu USB.

Napětí napájecích vodičů portu USB je $5V \pm 5\%$. USB hostitel poskytuje dle [16] proud 100 mA pro jedno zařízení. Jestliže zařízení požádá o navýšení proudu, může mu být poskytnut proud až 500 mA. Existují speciální porty USB, které poskytují proud až o velikosti jednotek ampér. Jedná se o tzv. „Charging ports“ [22], které jsou určeny pro zařízení s větším odběrem např. nabíječky.



Obr. 3.3: Obvod napájení.

Prvním článkem v obvodu napájení je sériově zapojená tlumivka, viz Obr.3.3. Tato cívka tvoří spolu s kapacitorem filtr druhého řádu typu dolní propust. Nežádoucí napěťové zákmity budou tímto filtrem odstraněny.

Dalšími prvky v obvodu napájení jsou srážecí diody a zkratovací propojka. Je-li zkratovací propojka odpojena, vzniká na diodách průtokem proudu úbytek napětí, který má za následek snížení výstupního napětí z 5 V na cca 3,5 V. Tento člen je zde zařazen z důvodu univerzality celého zařízení. Může se stát, že některé typy akumulátorů nebudou tolerovat sběrnici SMBus s logickými úrovněmi 5/0 V, ale pouze sběrnici s log. úrovněmi 3,6/0 V. Zkratovací propojkou lze nastavit napájecí napětí obvodů, tím i napětí logických úrovní.

3.4 Návrh plošného spoje

Deska plošného spoje byla navržena jako „jednostranná“ se čtyřmi drátovými propojkami. Rozložení součástek na desce je uvedeno v příloze A.2 a vrstva spojů je v příloze A.3. Při návrhu DPS je nezbytné dodržet několik zásad, týkajících se elektrické i konstrukční stránky zařízení.

Požadavky z elektrického hlediska:

- Kritické úseky vodičů by měly být co nejkratší. Mohou vyzařovat a rušit elmag. vlny, nebo se mohou naopak samy stát anténou elmag. vln. Největší problém může nastat ve spojení krystal – MCU a MCU – port USB.
- Dostatečně silné vodivé cesty, obzvláště v napájecím obvodu a zemnění. V tomto projektu je volná plocha DPS vyplněna zemním vodičem (GND). Tím je dosaženo snížení odporu GND. Standardní navržená tloušťka vodivých cest je 1,27 mm. V úsecích, kde je to nutné, je tloušťka cest mírně snížena.

Požadavky z konstrukčního hlediska:

- Již na počátku návrhu je výhodné naplánovat tvar a velikost DPS podle velikosti a tvaru boxu, do kterého má být zařízení usazeno. V případě tohoto projektu byl zvolen box s vnitřními rozměry 53 × 78 mm a se středovým sloupkem o průměru 70 mm. DPS tedy byla navržena v těchto rozměrech a se středovým otvorem.
- Je důležité rozvrhnout ukotvení DPS do boxu. Na stávající desce je realizováno ukotvení dvěma distančními sloupky. Tyto sloupky se pomocí šroubů ukotví do otvorů v rozích plošného spoje.
- Při návrhu je třeba zvolit vhodné rozložení součástek. Konektor USB se nachází na okraji desky. Otvorem v boxu se připojuje kabel USB do konektoru. Konektor pro sběrnici SMBus (přípojka Smart Battery) není součástí DPS, ale součástí boxu. Nachází se nad DPS. V oblasti, kde je tento konektor, se nemůžou nacházet vysoké součástky, které by zavazely, vyjma rezistorů a drátových propojek. Piny konektoru SMBus jsou v příloze A.2 uvedeny pod označením „Connector 1, 2, 3“.

4 SOFTWARE SYSTÉMU

Při výběru vhodných programovacích nástrojů byly vzneseny dva požadavky. Primárním požadavkem vývojových nástrojů je bezplatná licence (freeware, nebo Open Source). Druhým požadavkem je jejich dobrá dostupnost.

4.1 Program mikrokontroléru

Vývoj a ladění programu pro mikrokontrolér ATmega168 byl prováděn softwarem AVR Studio ver. 4.19. Tento software je spolu se všemi rozšiřujícími knihovnami a doplňky poskytován zdarma, a to výrobcem kontrolérů Atmel Corporation [23].

K vytvoření virtuálního hosta, tedy ke komunikaci s USB hostitelem, byl využit software V-USB[19], který je popsán v kap.2.5.1. Je to kód psaný v programovacím jazyce „C“. Je implementován do hlavního řídicího programu mikrokontroléru.

4.1.1 Komunikace na sběrnici SMBus

Z důvodu nekompatibility sběrnic I²C a SMBus nelze již nachystanou hardwarovou podporu sběrnice I²C v mikrokontroléru použít [2]. Je tedy nutné vytvořit softwarovou implementaci sběrnice SMBus. Tento fakt byl zjištěn až po detailním rozboru a porovnání sběrnic I²C a SMBus. Rozdíly sběrnic a důvody proč nemůže být použita hardwarová implementace I²C jsou detailně popsány v kap. 2.3.6.

Časování sběrnice je nastaveno ve shodě se standardem sběrnice SMBus [13]. Velikost časových kvant pro zpoždění, čekání či detekci, je navržena dle reálně naměřených hodnot při komunikaci s notebookovým akumulátorem DELL M911G. Tento akumulátor plně podporuje funkce stanovené standardem [13].

Každý proces komunikace začíná jedinečnou podmínkou „Start“ a končí jedinečnou podmínkou „Stop“:

```
void SMBstart(void) {           //START
    H_DAT;                      //nastavení SMB_DAT na „H“
    _delay_us(4);               //zpoždění
    H_CLK;                      //nastavení SMB_CLK na „H“
    _delay_us(7);               //zpoždění
    L_DAT;                      //nastavení SMB_DAT na „L“
    _delay_us(7);               //zpoždění - Thd:sta - min 4us
    L_CLK;                      //nastavení SMB_CLK na „L“
    _delay_us(10);              //zpoždění - Tlow - min 4,7us
}

void SMBstop(void) {           //STOP
    L_CLK;
    _delay_us(1);
    L_DAT;
    _delay_us(4);               //zpoždění - Tsu:dat - min 250 ns
    H_CLK;
    _delay_us(8);               //zpoždění - Tsu:sto - min 4us
    H_DAT;
```

```

    _delay_us(6);          //Tbuf - min 4,7us
}

```

Funkce vysílání jednoho bajtu (TX) má za úkol odesílat jednotlivé bity zprávy a přitom kontrolovat jejich příjem na druhém konci sběrnice. Zápis každého bitu doprovází podmínka potvrzení, že přijímač data přijal. Jakmile je na lince SMB_DAT připraven odchozí bit, přichází na linku SMB_CLK nástupná hrana. To znamená, že je hodinová linka uvolněna a zdvihnuta na logickou úroveň „H“. Toto je třeba vždy kontrolovat. Jestliže přijímací zařízení není připraveno, podrží hodinovou linku na úrovni „L“. Vysílač tedy musí vyčkat na zdvihnutí SMB_CLK, nebo přenos zrušit.

```

if(dataTX&0x80){          //vysílaný bit
    H_DAT; }
else{    L_DAT; }
dataTX<<=1;                //rotování out registru
_delay_us(10);             //Tsu:dat - min 250ns
H_CLK;                     //nástupná hrana hodinového cyklu
_delay_us(1);
if(bit_is_clear(SMB_PIN,SMB_CLK)) //korektní příjem
    _delay_us(5);
if(bit_is_clear(SMB_PIN,SMB_CLK)) //korektní příjem po 5 us
    _delay_us(10);
if(bit_is_clear(SMB_PIN,SMB_CLK)) //korektní příjem po 15 us
    ac++;                  //chybový registr +1
_delay_us(10);             //Thigh - min 4 us
L_CLK;                     //sestupná hrana CLK
}

```

Poslední částí odeslání osmi bitů je potvrzovací podmínka ACK. Na konci funkce TX je datová linka uvolněna a zjišťuje se potvrzení příjmu. To je signalizováno přijímačem tak, že změní datovou linku (v devátém bitu) na log „L“.

```

if(bit_is_clear(SMB_PIN,SMB_DAT)){
    ac=0;}                  //ACK - potvrzen příjem
else{
    ac=1;}                  //NACK - nepřijato

```

Příjem dat (RX) je obdobný, jako odesílání. Rozdíl je však v tom, že z datové linky SMB_DAT jsou data čtena. Příjem osmi bitů se provádí v osmi cyklech. Jeden průběh cyklu má následující charakter:

```

b<=<1;                    //rolování čtecího bufferu
do{                      //proběhne alespoň jednou
    H_CLK;                //nástupná hrana CLK - začíná čtení
    _delay_us(1); }       //krátké čekání
while(bit_is_clear(SMB_PIN,SMB_CLK)); //Slave není připraven
_delay_us(2);             //čtení uprostřed bitu
if(bit_is_set(SMB_PIN,SMB_DAT)){ //komparace
    b|=0B00000001;}      //Rx je „H“
else
    b&=0B11111110;       //Rx je „L“
_delay_us(3);             //prodloužení bitu na 5 us
L_CLK;                    //sestupná hrana CLK

```

Příjem a odesílání dat se ve většině Smart Battery příkazů provádí po dvou bajtech. Tyto protokoly se nazývají ReadWord a WriteWord, viz kap. 2.3.3. Čtení slova (16 b) se provádí dvakrát opakovaným čtením bajtu. Stejně tak je tomu i se zápisem 16b slova, viz Obr. 2.12 a Obr. 2.13.

4.1.2 Komunikace na sběrnici USB

Jak již bylo nastíněno v kap. 2.5.1, k vytvoření virtuálního hosta USB v mikrokontroléru je použit hotový kód V-USB [19], který je napsaný v programovacím jazyce C.

Zařízení se chová jako standardní HID zařízení (třída 3), které s hostitelem komunikuje pomocí primárního endpointu 0 a pomocí endpointu 3 [17]. Deskriptor je soubor dat, který popisuje chování interface USB. Je po připojení zařízení načten do PC. Deskriptor stanovuje např. délku paketu, velikost, směr dat, atd.

```
0x15, 0x00,          // logické minimum (0)
0x26, 0xff, 0x00,    // logické maximum (255)
0x75, 0x08,          // počet bitů v paketu (8)
0x95, 0x03,          // počet paketů (3)
...
```

Aktivitu na sběrnici USB monitoruje funkce „usbFunctionSetup“. Tato funkce zjišťuje, zdali se jedná o příchod dat na sběrnici a dává podněty ke zpracování:

```
usbRequest_t *rq = (void *)setupData; //přetypuje data pro analýzu
switch(rq->bRequest){
    case USBRQ_HID_SET_REPORT:          //jedná se o příchod dat
        currentPosition = 0;             //výchozí pozice pro čtení
        bytesRemaining = rq->wLength.word; //uloží délku
        if(bytesRemaining > sizeof(prijata_data))
            bytesRemaining = sizeof(prijata_data);
        //limituje data dle velikosti bufferu
        return USB_NO_MSG;               //nastaví příznak pro čtení
}
```

Příjem dat, resp. načtení dat do uživatelské paměti, se kterou se bude dále pracovat, je realizován funkcí „usbFunctionWrite“:

```
if(len > bytesRemaining) //limitace množství, které lze uložit
    len = bytesRemaining;
bytesRemaining -= len;    //zjištění zbylého místa pro uložení
prijata=1;               //příznak pro zpracování
for(i = 0; i < len; i++)
    prijata_data[currentPosition++] = data[i]; //načtení dat
return bytesRemaining == 0; //buffer je plný - hodnota 1
```

Během spojení hosta USB a hostitele musí být spojení neustále udržováno. Funkce „usbPoll“ udržuje spojení „na živote“. Jakmile je funkce vypnuta (zacyklení, zdržení

v podprogramu) spojení se přeruší. Největší interval mezi dvěma funkcemi „usbPoll“ byl experimentálně změřen a stanoven na cca 60 ms.

Při běhu programu je třeba nulovat resetovací čítač Watchdog. Tento čítač je v kontroléru ATmega168 implementován hardwarově. Jakmile čítač dosáhne vrcholu, resetuje mikrokontrolér. Při spuštění programu je nastaven na periodu 1 s. Watchdog řeší případné zacyklení programu, a to ať již ve zpracování, nebo při aktivitě na sběrnici USB.

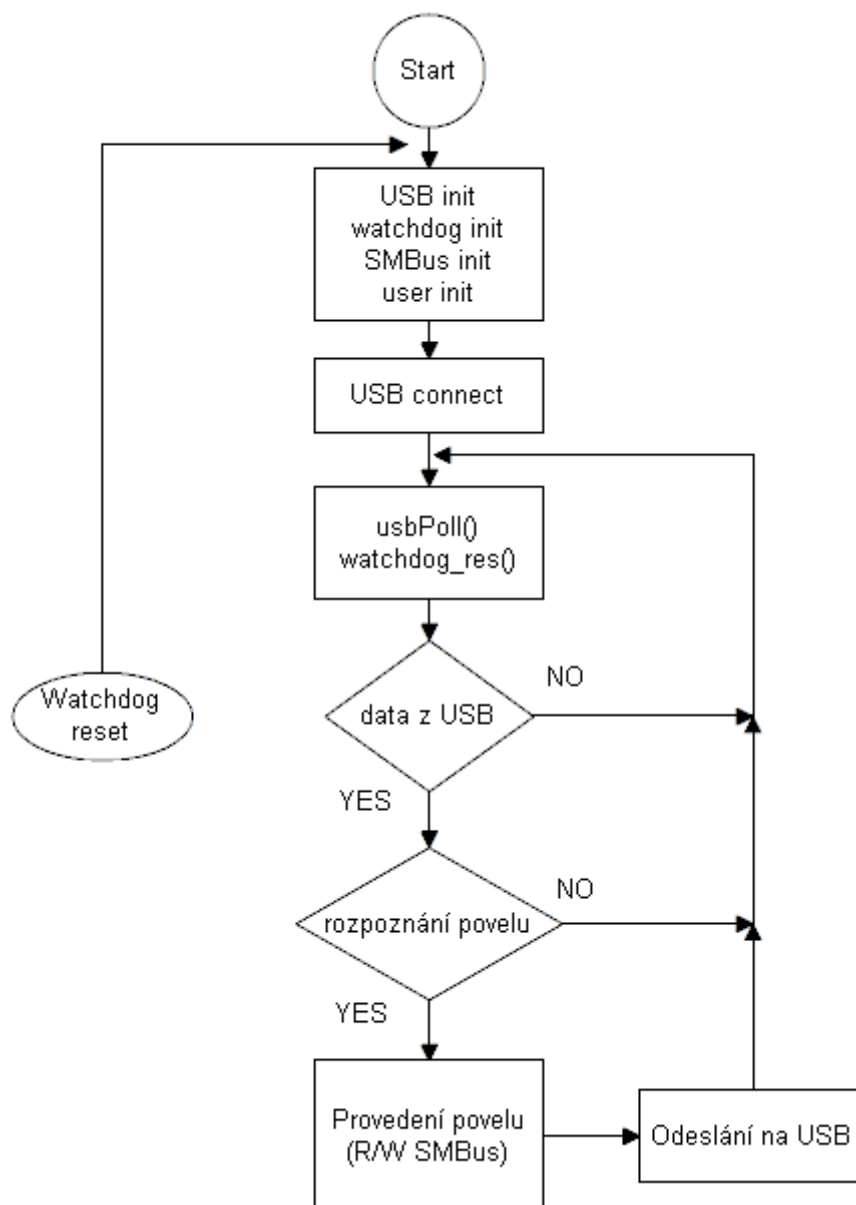
Funkcí, která má na starosti zpracování dat, je funkce „action“. Pomocí jí jsou rozpoznávány povely, které posílá uživatel z PC.

```
if(*rxd==48){          //uživatel chce zjistit adresu Smart Battery
    detect();          //zjištění adresy
    *txd=address1;     //uložení do bufferu
    send=1; }          //příznak pro odeslání
if(*rxd==49){          //uživatel chce použít vlastní adresu
    rxd++;             //posun ukazatele
    address1=*rxd; }   //uložení poslané adresy
if(*rxd==temp||*rxd==voltage||*rxd==cycle_count||*rxd==man_date){
    //příkaz pro čtení teploty, data výroby, atd...

    if(*txd==temp)
        ci=read_w(address1,temp);          //čtení teploty
    if(*txd==voltage)
        ci=read_w(address1,voltage);        //čtení napětí
    if(*txd==cycle_count)
        ci=read_w(address1,cycle_count);    //čtení počtu cyklů
    if(*txd==man_date)
        ci=read_w(address1,man_date);       //čtení data výroby
    txd++;
    *txd=ci&0xFF;      //rozložení na horní a dolní byte
    txd++;
    ci>>=8;
    ci&=0xFF;          //rozložení na horní a dolní byte
    *txd=ci;
    send=1; }          //příznak pro odeslání na USB
```

Na Obr.4.1 je vyobrazen vývojový diagram, dle kterého běží program mikrokontroléru. Po startu je provedena inicializace (USB, SMBus, porty, SFR, atd.). Následně se zařízení pokouší připojit k hostiteli USB. Toto spojení je udržováno příkazy „usbPoll“ a „watchdog_res“. Pokud se na zařízení nic neděje – žádná data nepřicházejí ani neodchází, běží tato krátká smyčka znovu dokola. Jakmile přijdou data z portu USB, jsou podrobena analýze. Jedná-li se o příkaz, je provedena operace dle tohoto příkazu (typicky čtení ze Smart Battery). Načtená data jsou nakonec odeslána přes sběrnici USB do PC a program se vrací zpět do nekonečné smyčky.

Dojde-li v některé části programu k zacyklení, zareaguje watchdog timer a vynuluje čítač programu – resetuje mikrokontrolér. Program pak běží od začátku.



Obr. 4.1: Vývojový diagram programu mikrokontroléru.

4.2 PC aplikace – Smart Battery Monitor

K vytvoření aplikace bylo využito vývojové prostředí Visual Studio 2008 [10]. Program byl napsán v programovacím jazyce C#. Primárním úkolem bylo nejprve zvolit a vyhledat vhodný ovladač portu USB, a potom zprovoznit přenos dat na sběrnici USB. Dalším krokem bylo napsání funkcí, které odesílají uživatelská data na linku USB a následně dekódují data příchozí. Tato data jsou posléze ukládána do tabulky a zobrazena uživateli. Uživatel má možnost tabulku dat uložit do počítače na HDD. Nakonec bylo nutné vytvořit přehledné grafické rozhraní v podobě Windows formuláře, které uživateli poskytuje jak základní, tak i pokročilé možnosti.

4.2.1 Připojení a provoz dat na sběrnici USB

Z několika málo variant ovladačů sběrnice USB, které lze dohledat na internetu, byl zvolen freeware ovladač USB_HID_driver [4]. Ovladač je pod licencí GNU a je tedy volně šiřitelný.

K ovládání portu USB jsou použity předem hotové funkce a události. Na začátku programu proběhne inicializace a zároveň jsou nadefinována identifikační čísla (VID, PID), podle kterých je zařízení rozpoznáno.

```
InitializeComponent();  
this.usb.ProductId = 0x05df;           //ID zařízení  
this.usb.VendorId = 0x16c0;           //ID výrobce
```

Připojí-li se do portu USB zařízení s identifikátory VID a PID, automaticky se vyvolá příslušná událost. Odpojí-li se zařízení, je vyvolána jiná událost.

```
private void usb_OnSpecifiedDeviceArrived(...) { //připojení  
}  
  
private void usb_OnSpecifiedDeviceRemoved(...) { //odpojení  
}
```

Data připravená k odeslání se shromažďují v bufferu. Při přenosu je toto pole dat přeneseno najednou. Přenos je tedy realizován po blocích. V tomto projektu je velikost jednoho datového bloku 4 bajty. V prvním bajtu je vždy nula, která se do uživatelských dat nezapočítává. V druhém bajtu je příkaz (command) pro Smart Battery (rok výroby, napětí, ...), ve třetím a čtvrtém bajtu je hodnota, která má být do Smart Battery zapsána. Jedná-li se o čtení z akumulátoru, jsou poslední dva bajty nulové. Experimentálně bylo zjištěno, že na počátku přenášeného pole musí být vždy nula. V ostatních případech nedojde na straně mikrokontroléru k přijetí dat. Mikrokontrolér tuto nulu na začátku zprávy přeskakuje, prvním příchozím znakem se jeví „příkaz“ (command). K vlastnímu odeslání dat je použita hotová funkce:

```
usb.SpecifiedDevice.SendData(buffer_4B); //odeslání bloku dat
```

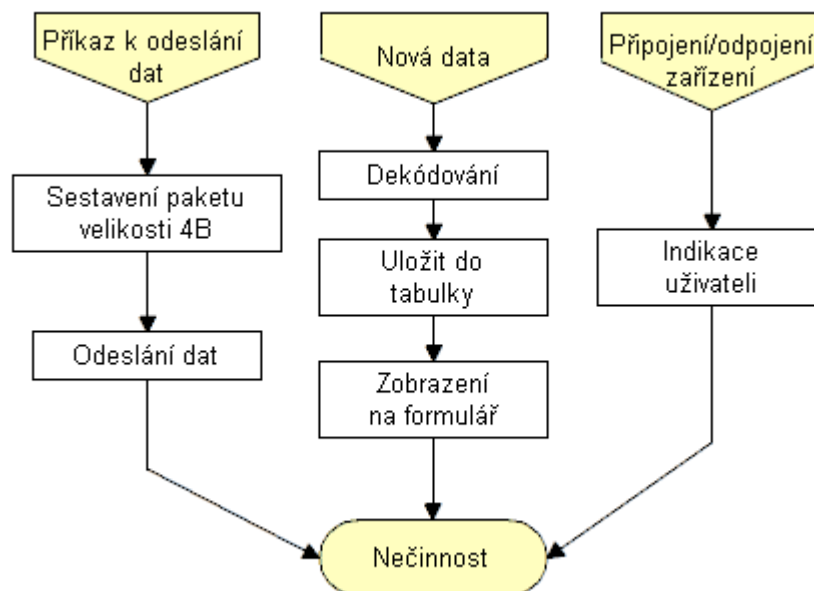
Příjem dat je signalizován spuštěním události. Přichází 3 bajty dat, které jsou uloženy v dočasné proměnné args. V prvním bajtu je příkaz, ve druhém a třetím bajtu je 16bitová hodnota. Tato data se dekodují a zpracují.

```
private void usb_OnDataRecieved(object sender,  
DataRecievedEventArgs args) {  
  
    cis1 = (uint)args.data[1];           //načtení dat  
    cis2 = (uint)args.data[2];  
    cis3 = (uint)args.data[3];  
    if(cis1 == 8) {                       //informace o teplotě  
        ...                               //zpracování, zobrazení  
    }
```

```

if(cis1 == 9){                                     //informace o napětí
    ...      }                                     //zpracování, zobrazení
}

```



Obr. 4.2: Vývojový diagram datové části aplikace.

Vývojový diagram na Obr.4.2 představuje zevrubné schéma, kterým je popsána „datová“ část aplikace, která má na starosti tok dat (příjem, zpracování, odeslání). Aplikace obstarává dvě hlavní události (příchod dat, připojení/odpojení) a jednu funkci (odeslání dat), viz Obr.4.2. Narozdíl od mikrokontroléru zde nemusí běžet nekonečná smyčka. Přerušení v podobě vyvolané události nastartuje procesy, které vždy končí přechodem do nečinnosti programu.

4.2.2 GUI – uživatelské rozhraní

Posledním krokem bylo vytvoření grafického rozhraní mezi uživatelem a aplikací. Pomocí grafických objektů platformy .NET [1] (tlačítek, editačních oken, panelů a obrázků) byl sestaven formulář, který slouží k pohodlnému ovládání funkcí.

Formulář aplikace lze rozdělit na tři části:

- Základní přehled – poskytuje uživateli základní přehled stavu Smart Battery. Nachází se pod záložkou „Overview“. Tlačítkem „Read“ lze načíst veškerá data z připojeného akumulátoru, tlačítkem „Save list“ lze tato data uložit na HDD. To vše lze vidět na obrázku v příloze B.1.
- Rozšířené možnosti – tato záložka s názvem „Advanced“ poskytuje uživateli nadstandardní funkce. Náhled je zobrazen v příloze B.2. Pomocí tlačítka „Send“ v příslušné sekci je možné číst/zapisovat 16bitové příkazy jednotlivě, nebo číst delší bloky dat. Veškerá zobrazená nebo zadávaná čísla jsou ve formátu desítkových čísel. Rozsah čísel je tedy 0 – 255 (8 bitů).
V rámci záložky rozšířených možností se dále nachází sekce pro správu

adresy přístroje (Address), se kterým hodlá uživatel komunikovat. Zařízení lze připojit i ke sběrnici, na které se nachází jiné zařízení než Smart Battery, přístroj s jinou adresou. Standardně by mělo každé zařízení Smart Battery mít adresu 11. Tlačítka „Use“ a „Detect“ lze buď nastavit předem známou adresu, nebo neznámou adresu detekovat. Touto možností se stává „Smart Battery Monitor“ univerzálnější.

- Status/režim – v této sekci se nachází indikátory stavu přenosu a stavu režimu přenosu. Sekce status/režim se nachází nahoře na formuláři, jak je vidět v příloze B.1. i B.2. Indikátory s názvem „USB“ a „SMBus“ ukazují uživateli, kde se při přenosu dat stala chyba, popř. zdali vůbec přenos probíhá. Zaškrtnutím pole „Autodetect“ se spustí kontinuální hledání připojeného zařízení. To lze s výhodou využít ke zjištění správnosti zapojení sběrnice na akumulátor (CLK, DAT, GND). Zatřením pole „CRC8“ lze vynutit potvrzení přenesených dat kontrolním součtem.

Na Obr.4.3. jsou zobrazeny fáze běhu programu v rámci uživatelského rozhraní. Vstupem je vždy aktivita vyvolaná uživatelem, samočinná aktivita (časovač), nebo jiná externí událost (příchod dat, výpadek, ...).

Přichází-li data, jsou zpracována. Pokud jsou tato data poškozena, nebo neúplná, je uživatel upozorněn chybovou hláškou. V opačném případě je výsledek, tedy informace o akumulátoru, přehledně zobrazen na formuláři.

Jestliže uživatel požaduje vlastní sestavení protokolu, použije tlačítko „Send“. V takovém případě je nutná korektura vstupních (uživatelských) znaků. Pokud je vše v pořádku a vstupní znaky spadají do definičního oboru příkazu, je sestaven uživatelský paket a posléze odeslán. Správnost vstupních dat ověřuje funkce TryParse:

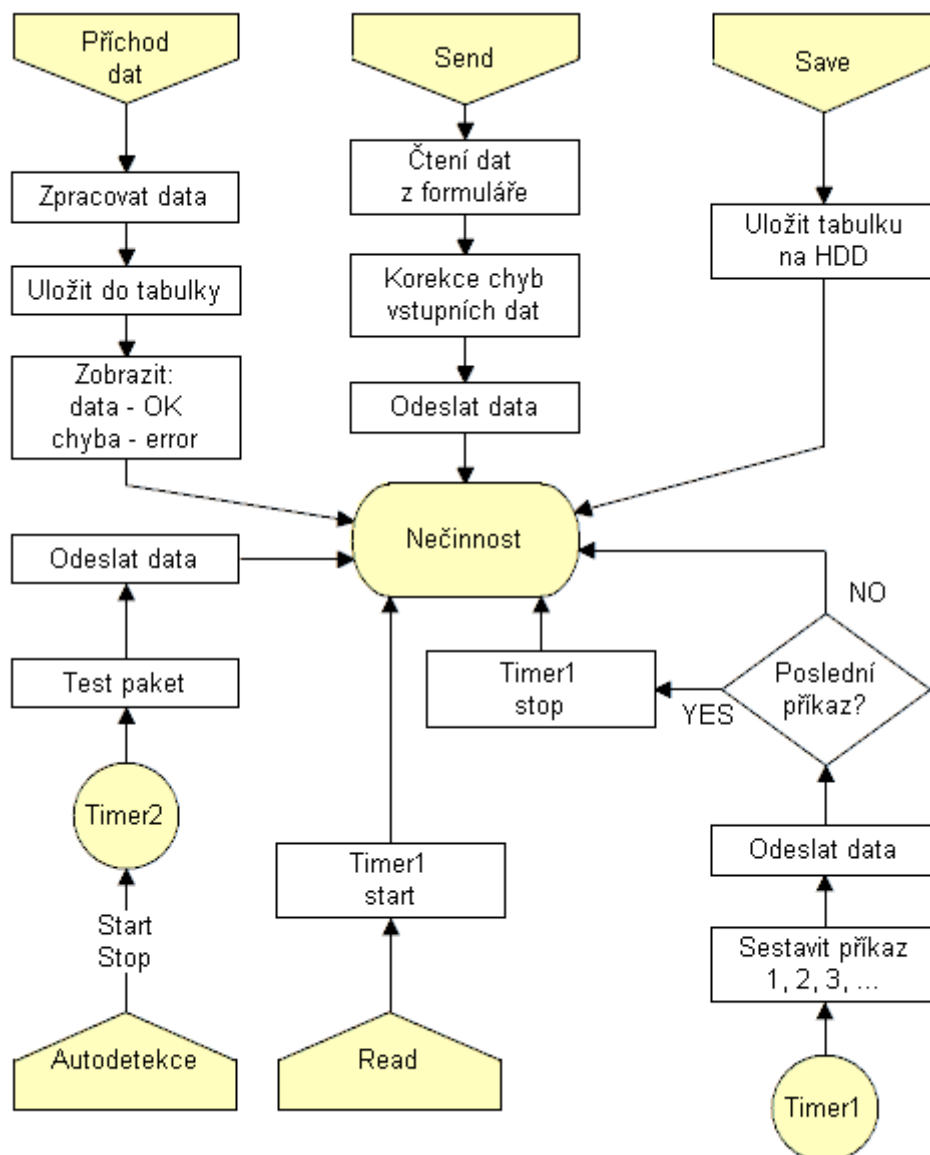
```
bool resultBool = int.TryParse(textBox5.Text, out converted);
if (resultBool == true){
    ...}                                //platný vstup textBox5.text
else{
    MessageBox.Show(„Error“);           //neplatný vstup textBox5.text
}
```

Spuštění autodetekce znamená spuštění časovače č. 2. Časovač periodicky odesílá testovací paket na sběrnici SMBus. Následně je očekávána odpověď. Nepřijde-li žádná odpověď do příštího taktu timeru č. 2, je uživateli nahlášena chyba. Je-li na sběrnici SMBus připojeno zařízení s libovolnou adresou, mělo by na dotazovací paket vždy odpovědět.

Načtení všech parametrů se spouští vstupem (tlačítkem) „Read“. Je odstartován Časovač č. 1, který sestavuje pakety s příkazy a postupně je odesílá na Smart Battery. Nakonec se po odeslání posledního příkazu sám zastaví. Během asi dvousekundového intervalu jsou z akumulátoru vyčtena uživatelská data. Čas poslání následujícího paketu nezávisí na příjmu odpovědi z předchozího paketu. Interval dvou po sobě jdoucích paketů (dotazů), je dán pevně, a to 250 milisekund. Takto je timer nastaven ze dvou důvodů:

- Některé akumulátory nepodporují kompletní sadu příkazů, a tudíž by na jim neznámý dotaz neodpověděly – čtení by se zastavilo.

- Čtení bloku dat trvá déle. Blok dat je rozdělen na malé subpakety po 3 bytech. Přicházejí z MCU jeden po druhém a vždy další dva znaky se přičtou k předchozím. Není nikdy jasno, kdy bude blok (výrobce, název, chemie,...) kompletní.



Obr. 4.3: Vývojový diagram grafického rozhraní aplikace

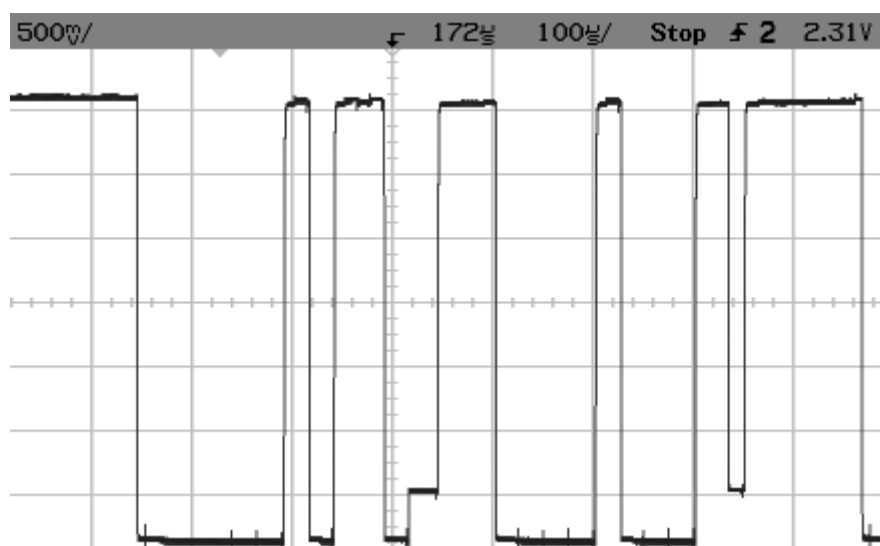
Posledním vstupem vývojového diagramu na Obr 4.3 je „Save“. Po načtení několika akumulátorů vzniká seznam/kontejner naplněný objekty – tedy akumulátory. Tlačítkem „Save“ je možné vyvolat dialogové okno, které provází uživatele. Toto dialogové okno typu „saveFileDialog“ nabízí uživateli pojmenování a uložení dat do textového souboru.

5 TESTOVÁNÍ

5.1 Komunikace na sběrnici SMBus

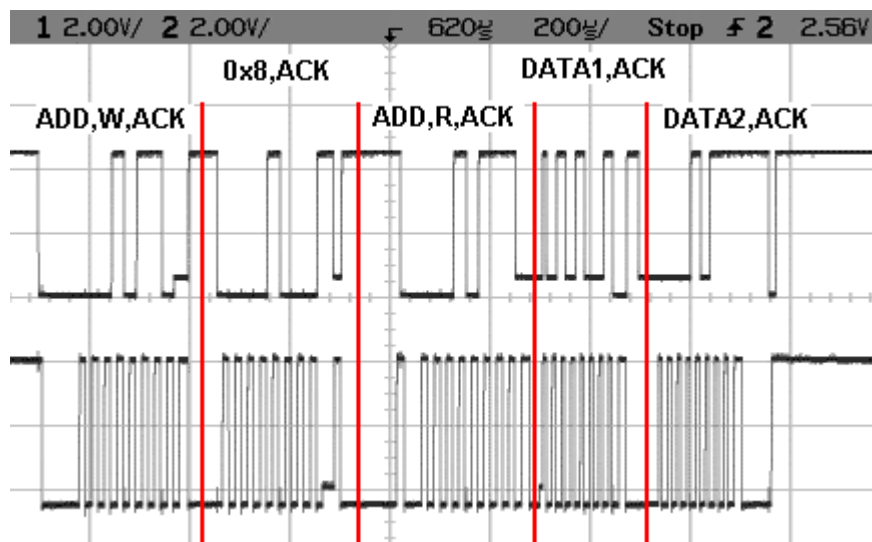
V reálném světě je vždy kladen velký důraz na minimální spotřebu energie. Podobně je tomu tak i na poli baterií. Ve standardu Smart Battery je dán maximální proud I_{PULLUP} , který teče zdvihacími rezistory při úrovni „L“, viz Tab. 2.5. Existence tohoto limitu ovlivňuje jmenovitou hodnotu odporu těchto rezistorů. Snížení proudu I_{PULLUP} znamená zvýšení odporu rezistorů, což má za následek zaoblení nástupných hran na sběrnici.

V zařízení „Interface diagnostické sběrnice pro akumulátory“ není limit I_{PULLUP} dodržen. Napájení je vzato z portu USB. V tomto případě tedy nedochází k úbytku energie z akumulátoru vůbec. Je zbytečné kvůli potřebě I_{PULLUP} deformovat signál a tím zvyšovat pravděpodobnost chybně detekované úrovně. Zvolené rezistory mají odpor $1500\ \Omega$ a teče jimi proud $3,3\ \text{mA}$ (resp. $2,4\ \text{mA}$ při $V_{CC} = 3,6\text{V}$). Ostrost hran datového signálu je viditelná na Obr.5.1, který byl pořízen při tetování.



Obr. 5.1: Provoz na datové lince SMB_DAT.

Vstupně-výstupní piny Smart Battery jsou pravděpodobně realizovány jako typ otevřený kolektor. Na linku musí být připojeny zdvihací rezistory. Je-li potřeba na linku poslat logickou úroveň „L“, otevře se tranzistor, který zkratuje SMB_DAT se zemním vodičem. Na výstupu je pak logická úroveň „L“ zvýšená o úbytek napětí mezi kolektorem a emitorem (U_{CE} resp. U_{DS}) řízeného tranzistoru. Toto zvýšení napětí je velmi zřetelně vidět na Obr. 5.1.



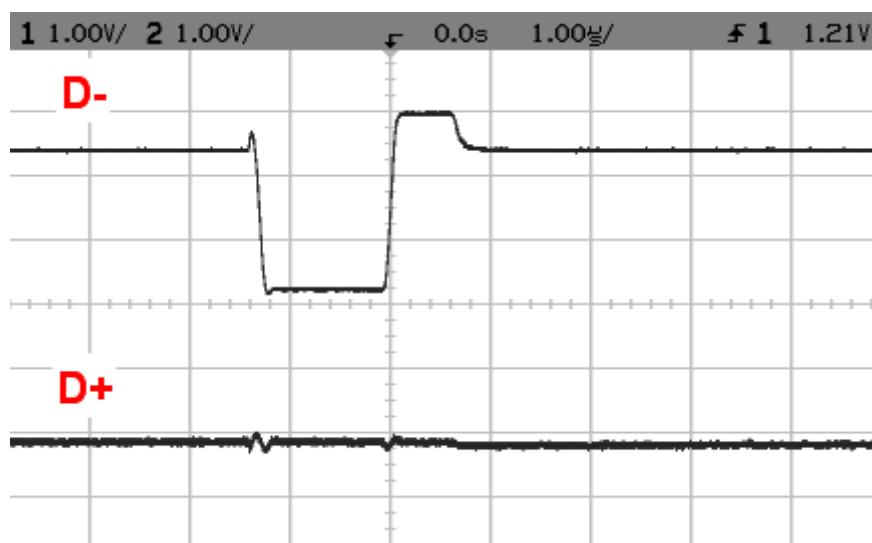
Obr. 5.2: SMBus komunikace.

Jak v reálu vypadá přenos dat na sběrnici SMBus ukazuje Obr.5.2. Zde je ukázán protokol „Read Word“, kterým se ze Smart Battery načte 16bitové slovo. Konkrétně se jedná o příkaz pro načtení teploty (číslo 0x08 hexadecimálně).

V první části se adresuje „Slave“ a určí se směr dat (write). V druhé části se stanoví, jaká data mají být načtena. Třetí částí je opět adresování „Slave“ a určení směru toku dat (read). Ve čtvrté a páté části Obr.5.2 již probíhá vlastní stahování dat. Každý blok osmice symbolů je potvrzen symbolem ACK.

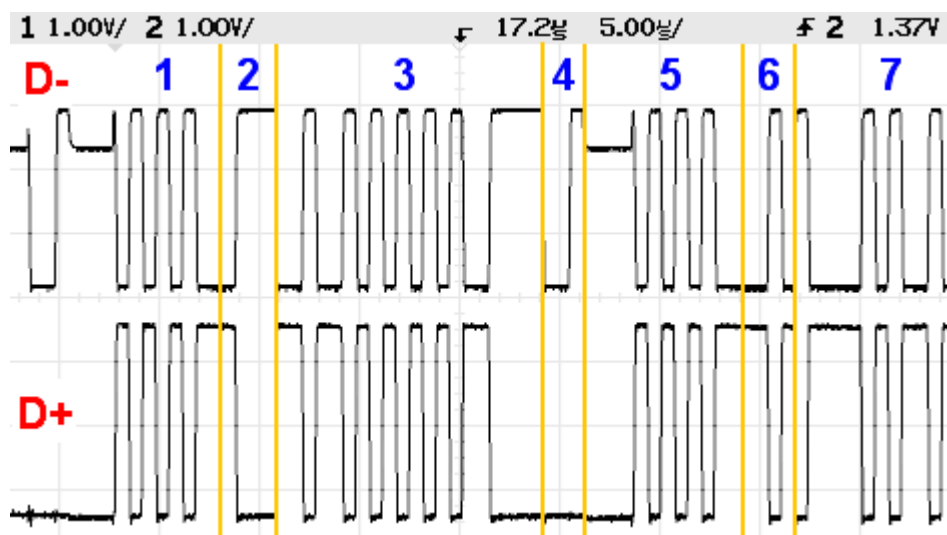
5.2 Komunikace na sběrnici USB

Jak již bylo řečeno výše, spojení hostitele USB a hosta musí být periodicky aktualizováno. Podněty k aktualizaci spojení vyvolává host, a to voláním funkce `usbPoll()` v hlavní smyčce programu. Na Obr. 5.3 je zobrazen časový průběh pulzu, kterým dává host hostiteli najevo, že je stále přítomen a spojení trvá.



Obr. 5.3: Periodická indikace „usbPoll()“.

Při testování byl zachycen i průběh toku dat na sběrnici USB. Na Obr. 5.4 je zobrazen náhodný časový průběh přenosu jednoho paketu a část následujícího paketu. Data jsou kódována diferenčně (NRZI). Jedná-li se o čtyřbitové pole PID, přichází jako první MSB. Pole datové přichází reverzně – první je LSB. Po dekódování dat z Obr. 5.4 bylo zjištěno, že se jedná o datový paket (2), který je charakterizován specifickým PID. Druhým paketem je paket speciální (6).



Obr. 5.4: Provoz na sběrnici USB.

Časový průběh na Obr.5.4 je pro přehlednost rozdělen do několika částí:

- 1) Synchronizační sekvence.
- 2) Paket ID – 4 symboly.
- 3) Data – n symbolů.
- 4) EOP – konec paketu – 3 symboly.
- 5) Synchronizace druhého paketu.
- 6) Paket ID – 4 symboly.
- 7) Data – n symbolů.

5.3 Univerzálnost zařízení

Při realizaci zařízení, ať již hardware nebo software, bylo přihlíženo na univerzálnost, praktičnost a všestranné použití tohoto zařízení. V praxi to znamená, že by toto zařízení mělo zvládnout komunikaci s různými akumulátory, které podporují, nebo alespoň napodobují standard Smart Battery.

Logické úrovně na sběrnici SMBus je možné změnit z 5/0 V na 3,6/0 V. Časování linky SMBus je adaptivní, což je u pomalejších akumulátorů nutné (jde o prodloužení periody hodinového signálu). Je zavedeno čekání na opožděné potvrzení dat (ACK), které je způsobeno zpracováním předešlých dat.

Zařízení bylo připojeno k několika typům notebookových akumulátorů a byla navázána komunikace. V Tab. 5.1 jsou tyto testované akumulátory uvedeny.

Tab. 5.1: Testované akumulátory

Výrobce	Typ	Funkčnost
Dell	M911G	Plná podpora
Toshiba	PA3536U-1BRS	Částečná podpora
Asus	A32-F3	Plná podpora

Akumulátor Toshiba PA3536U-1BRS měla podporu jen některých instrukčních zpráv. Z měření na sběrnici SMBus vyplývá, že v komunikaci kontroléru problém není. Problém je s největší pravděpodobností v akumulátoru. Čip uvnitř totiž nepodporuje některé zprávy standardu Smart Battery.

5.4 Konektor akumulátorů Smart Battery

Vzhledem k počtu výrobců elektroniky na trhu je potřeba počítat i s velkou rozmanitostí typů akumulátorů. Obecně lze říci, že každý výrobce přenosného počítače (každá značka) má jiný napájecí akumulátor, a to z pohledu napětí, kapacity, tvaru, atd. Někteří výrobci mají použity odlišné typy akumulátorů dokonce i pro každou produktovou řadu počítačů. Jediným společným rysem, který se ustálil ve vývoji notebookových akumulátorů, je typický konektor. Většina výrobců používá konektor s nožovými kontakty a bočními patkami, jak je vidět na Obr.5.5. Konektor může obsahovat více či méně kontaktů. Standard Smart Battery našel největší uplatnění právě u přenosných počítačů. Není však zaměřen pouze na PC, ale všeobecně na napájení mobilních zařízení, např. telefonů, kamer, atd. [12] U těchto menších zařízení bude pravděpodobně použit konektor jiný – menší.

5.4.1 Akumulátor DELL M911G

Jeden z široké řady notebookových akumulátorů výrobce DELL je typ M911. Parametry jsou uvedeny v Tab. 5.2. Připojení k PC je realizováno nožovým konektorem o devíti pinech, který je znázorněn na Obr.5.5. Devět nožů na straně počítače se zasouvá do oněch devíti kontaktů na konektoru. Konektor je ze čtyř stran usazen do obalu a ze dvou stran (zepředu a shora) je možné se k němu připojit. Záleží jen na stylu uchycení akumulátoru v přenosném počítači. Obr.5.5 znázorňuje popis kontaktů 1 – 9 z čelního pohledu, tedy z pohledu rovnoběžného se směrem zasouvání nožů.

Tab. 5.2: Parametry testovaného akumulátoru

Název	Typ	Kapacita	Napětí	Chemie
DELL	M911G	41 Wh	11,1 V	Li-Ion

Experimentálně bylo zjištěno zapojení konkrétních pinů na konektoru tohoto typu akumulátoru. Označení koresponduje s Obr.5.5:

1. GND – nulový potenciál celého obvodu, napájení.
2. GND, shodné s 1.
3. NC – bez funkce.
4. 2,8 V – ovládání napájecího výstupu VCC (pull-up rezistor).

5. GND – indikace přítomnosti akumulátoru v PC.
6. 0 V – SMB_DAT.
7. 0 V – SMB_CLK.
8. 1,8 V – VCC/nabíjení.
9. 1,8 V – VCC/nabíjení.



Obr. 5.5: Konektor akumulátoru – čelní pohled

Napájecí kontakty (GND a VCC) jsou zdvojeny, a tím je zvýšena jejich proudová zatížitelnost. Nalézají se na opačné straně konektoru. Protěžší konektor (na PC) má tyto napájecí kontakty prodloužené tak, aby při zapojování akumulátoru k počítači došlo nejprve k dotyku napájecích vodičů, a pak teprve k dotyku ovládacích vodičů. Zabrání se tak výskytu náhodných přechodových stavů.

Je-li akumulátor odpojený od PC, jsou kontakty 8 a 9 z důvodu bezpečnosti odpojeny. To se projeví snížením napětí na cca 2 V. Nenulová úroveň je pravděpodobně zapříčiněna úbytky napětí na interních obvodech. Počítač dává najevo svou přítomnost akumulátoru tím, že připojí kontakt č. 4 na nulový potenciál (GND). Tím se zapojí kontakt 8 a 9 ke zdroji. Naopak přítomnost akumulátoru je počítači signalizována kontaktem 5. Ten, jako v předchozím případě kontakt č. 4, spojí příslušný výstup z PC k potenciálu GND. Takto je realizována obousměrná signalizace přítomnosti akumulátoru. Zapojení a veškeré vlastnosti akumulátoru M911G byly zjištěny experimentálně.

5.4.2 Příklady některých typů akumulátorů

V tabulce Tab.5.3 je uveden soupis několika typů akumulátoru, které se na trhu vyskytují a jejich zapojení konektoru. Některé z nich byly vyzkoušeny experimentálně, některé jsou již odzkoušeny a je možné jejich zapojení dohledat v literatuře.

Tab. 5.3: Zapojení konektoru na Smart Battery

Značka	Typ	Počet kontaktů	Kontakt GND	Kontakt CLK	Kontakt DAT	Zdroj
DELL	M911G	9	1, 2	7	6	Exp.
DELL	3R305	9	8, 9	3	4	[3]
Toshiba	PA3536U-1BRS	9	8, 9	3	4	Exp.
IBM	-	6	6	3	4	[7]
HP	EV088AA	6	1	4	5	Exp.

Exp... bylo experimentálně zjištěno.

System „Interface diagnostické sběrnice pro akumulátory“ se k měřenému akumulátoru připojuje pomocí sběrnice SMBus. To znamená, že musí být na konektoru nalezen vodič GND, SMB_CLK a SMB_DAT. Jestliže uživatel neví, jak je konektor zapojen, musí náhodně vyzkoušet všechny možné kombinace kontaktů na konektoru. Počet možných kombinací je pak výsledek součinu permutace [11] a kombinace [9]:

$$M = C_k(n) \cdot P(k), \quad (5.1)$$

přičemž platí

$$P(k) = k! \quad (5.2)$$

a

$$C_k = \frac{n!}{k!(n-k)!}, \quad (5.3)$$

kde: k...počet přípojných kontaktů (3 kontakty na sběrnici SMBus),
n...počet kontaktů na konektoru,
M...počet možných kombinací,
C...kombinace [9],
P...permutace [11].

Například při zapojování konektoru akumulátoru DELL M911G (3 kontakty z 9) existuje až 504 různých kombinací. Toto náhodné hledání sběrnice SMBus je v praxi zcela nemožné. Proto musí být zavedeny zjednodušující podmínky.

Na několika akumulátorech bylo experimentálně zjištěno, že se napájecí kontakty vždy nacházejí v páru (proudové posílení) na okraji konektoru. Dalším zjednodušením je fakt, že se kontakty SMB_CLK a SMB_DAT u testovaných akumulátorů nacházely vždy vedle sebe. Tím odpadá kombinace náhodných nesousedících kontaktů a počet pokusů se velice zredukuje. Je-li navíc splněna podmínka, že uživatel našel zemní kontakt GND (např. pomocí voltmetru), počet kombinací je pak dán vztahem

$$M = (n-5) \cdot P(2), \quad (5.4)$$

který vychází z rovnice (5.1). Je platný za výše zmíněných podmínek.

Za těchto podmínek existuje při zapojení akumulátoru DELL M911G (9 kontaktů na konektoru) maximálně 8 různých kombinací. Toto číslo je již pro uživatele přijatelnější. S použitím automatického hledání (detekce) akumulátoru, viz kap.4.2.2, je úloha nalezení sběrnice na konektoru otázkou několika desítek sekund.

Rovnici (5.4) je třeba brát s určitou rezervou. Vychází z několika typů experimentálně ověřených akumulátorů Smart Battery.

ZÁVĚR

V této práci bylo zkonstruováno diagnostické zařízení pro notebookové akumulátory, které bylo po sléze i otestováno. Všechny kroky od začátku (teor. rozbor, návrh, programování, ...) k tomuto cíli směřovaly. Na počátku byl diskutován a navrhnut koncept zařízení. Po sléze bylo navrženo blokové schéma, typ kontroléru, SW prostředky a byly rozebrány sběrnice I²C, SMBus a USB. Přes tyto sběrnice spolu jednotlivé bloky zařízení komunikují.

V dalším kroku byly rozebrány a popsány parazitní jevy, které mohou při běžném provozu nastat. Při návrhu zapojení byl na tyto jevy, jako je např. ESD výboj, brán ohled. Druhým krokem návrhu byla deska plošného spoje. Tato deska měla předem dané rozměry, a to dle zvoleného krytí – boxu, do kterého byla později usazena. Dle tohoto návrhu byla deska i sestrojena, osazena součástkami a oživena.

V návaznosti na oživení desky plošného spoje byl vyvinut software mikrokontroléru, který komunikuje jak s akumulátorem, tak i s PC. Bylo zjištěno, že hardwarová implementace sběrnice I²C, kterou nabízí zvolený mikrokontrolér, nelze využít. Proto je komunikace s akumulátorem nakonec realizována ryze softwarově. Dále byla vyvinuta aplikace pro zpracování dat na PC. Tato aplikace řídí celý systém, zobrazuje data uživateli a ukládá je na harddisk počítače. Veškeré programové vybavení, které bylo použito, je dostupné pod licencí freeware.

Nakonec byl proveden test. Diagnostické zařízení bylo podrobeno zkoušce funkčnosti na třech rozdílných akumulátorech. Dva fungovaly bezproblémově, třetí akumulátor podporoval pouze některé instrukční zprávy. Výsledek měření říká, že v komunikaci problém není. Je tedy velmi pravděpodobné, že se problém nachází na straně akumulátoru, a to problém s podporou standardu Smart Battery.

Rozbor konektoru notebookového akumulátoru v samém závěru práce je sice mimo rámec zadání, ale je přáním zadavatele projektu. Je zde předložen detailní popis konektoru na vybraném typu akumulátoru značky DELL. A nakonec je stanoven doporučený postup při zapojování podobných konektorů na ostatních akumulátorech.

Zkonstruované zařízení může nalézt velmi dobré uplatnění všude tam, kde je zapotřebí rozhodovat o stavu akumulátorů typu Smart Battery. Právě místo by toto zařízení našlo ve zkušebnách, na oddělení reklamací a v opravárnách notebooků.

LITERATURA

- [1] .NET. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2012-2-16 [cit. 2012-04-23]. Dostupné z: <http://cs.wikipedia.org/wiki/.NET>
- [2] ATMEL CORPORATION. *ATmega168: Datasheet* [online]. rev.2545T-AVR-05/11. 2011 [cit. 2012-11-21]. Dostupné z: <http://www.atmel.com/Images/doc2545.pdf>
- [3] DELL D500 D600 battery Connector Pinout - AllPinouts. In: *AllPinouts* [online]. 2009 [cit. 2013-05-06]. Dostupné z: http://www.allpinouts.org/index.php/DELL_D500_D600_battery
- [4] FLORIAN LEITNER - FISCHER. *Usb HID Driver Library* [online]. 2007 [cit. 2012-12-06]. Dostupné z: <http://www.florian-leitner.de/index.php/projects/usb-hid-driver-library/>
- [5] GAJDOŠ, M. *Univerzální analyzátor sériových sběrnic*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 85 s. Vedoucí diplomové práce Ing. Petr Sysel, Ph.D.
- [6] HRBÁČEK, J. *Komunikace mikrokontroléru s okolím: 1. díl*. Praha: Nakladatelství BEN, 1999. ISBN 80-86056-42-2.
- [7] IBM ThinkPad Battery Connector Pinout - AllPinouts. In: *AllPinouts* [online]. 2008-4-28 [cit. 2013-05-09]. Dostupné z: http://www.allpinouts.org/index.php/IBM_ThinkPad_Battery
- [8] I²C. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2012-3-5 [cit. 2012-03-23]. Dostupné z: <http://cs.wikipedia.org/wiki/I%C2%B2C>
- [9] Kombinace. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2013-3-9 [cit. 2013-05-08]. Dostupné z: <http://cs.wikipedia.org/wiki/Kombinace>
- [10] Microsoft Corporation: Download Center. *Microsoft Visual Studio 2008 Service Pack 1 (Installer)* [online]. 2008-11-8 [cit. 2012-04-25]. Dostupné z: <http://www.microsoft.com/download/en/details.aspx?id=13276>
- [11] Permutace. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2013-3-9 [cit. 2013-05-08]. Dostupné z: <http://cs.wikipedia.org/wiki/Permutace>
- [12] System Management Interface Forum, Inc. *Smart Battery Data Specification* [online]. [cit. 2012-2-28]. Dostupné z: <http://sbs-forum.org/specs/sbdat110.pdf>
- [13] System Management Interface Forum, Inc. *System Management Bus (SMBus) Specification* [online]. Version 2.0. 2000-8-3 [cit. 2012-3-23]. Dostupné z: <http://www.smbus.org/specs/smbus20.pdf>
- [14] System Management Bus. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2012-2-12 [cit. 2012-03-26]. Dostupné z: http://en.wikipedia.org/wiki/System_Management_Bus
- [15] Texas Instruments Inc. *TMP101: Datasheet* [online]. [cit. 2012-3-1]. Dostupné z: <http://www.ti.com/lit/ds/symlink/tmp101.pdf>
- [16] Universal Serial Bus. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001 [cit. 2012-04-18]. Dostupné z: http://en.wikipedia.org/wiki/Universal_Serial_Bus

- [17] USB in a NutShell. In: *Beyond Logic* [online]. 2010-9-17 [cit. 2012-04-21]. Dostupné z: <http://www.beyondlogic.org/usbnutshell/usb3.shtml>
- [18] Using the I²C Bus. *Robot Electronics* [online]. England, 2001 [cit. 2012-04-25]. Dostupné z: http://www.robot-electronics.co.uk/acatalog/I2C_Tutorial.html
- [19] V-USB: Virtual USB port for AVR® microcontrollers. OBJECTIVE DEVELOPMENT SOFTWARE GMBH. *Objective Development* [online]. 2012 [cit. 2012-03-26]. Dostupné z: <http://www.obdev.at/products/vusb/index.html>
- [20] Fundamentals of Electrostatic Discharge Part One—An Introduction to ESD. In: *Electrostatic Discharge Association* [online]. 2010 [cit. 2012-11-16]. Dostupné z: <http://esda.org/fundamentalsP1.html>
- [21] Vratná pojistka. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2012-10-25 [cit. 2012-11-21]. Dostupné z: http://cs.wikipedia.org/wiki/Vratn%C3%A1_pojistka
- [22] USB IMPLEMENTERS FORUM, Inc. *Battery Charging v1.2 Spec and Adopters Agreement* [online]. 2010-12-07 [cit. 2012-11-24]. Dostupné z: http://www.usb.org/developers/devclass_docs/BCv1.2_070312.zip
- [23] Studio Archive. *ATMEL CORPORATION* [online]. 2011 [cit. 2012-12-03]. Dostupné z: <http://www.atmel.com/tools/studioarchive.aspx>

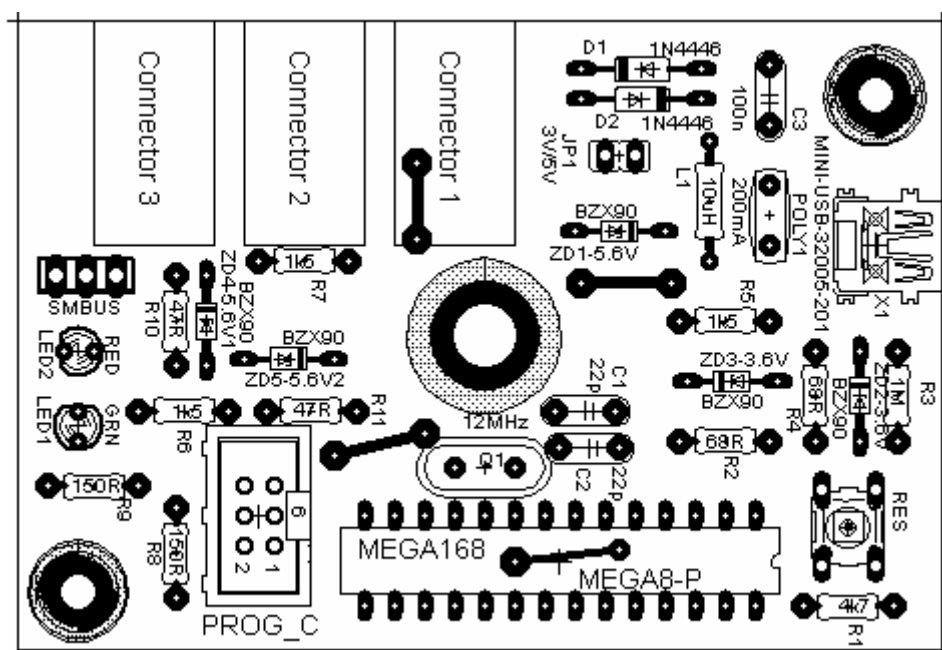
SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

AVR	Označení rodiny 8bitových mikrokontrolérů
BIN	Binární hodnota
DPS	Deska plošného spoje
ESD	Elektrostatický výboj
GND	Nulový potenciál/vodič nulového potenciálu
GNU	Projekt, který je zaměřený na svobodný software
H	Vysoká logická úroveň
HDD	Pevný disk
HW	Hardware (fyzicky existující vybavení přístroje)
I ² C	Dvoulinková sériová sběrnice
ISP	Sériové programování MCU
L	Nízká logická úroveň
MCU	Mikrokontrolér, mikroprocesor
OS	Operační systém
RISC	Procesory s redukovanou instrukční sadou
SFR	Speciální funkční registry
Smart Battery	Baterie obsahující speciální HW pro řízení a správu
SMB_CLK	Taktovací signál/vodič
SMB_DAT	Datový signál/vodič
SMBus	Sběrnice pro komunikaci Smart Battery s ostatními zařízeními
USB	Univerzální sériová sběrnice
X	Libovolná logická úroveň

SEZNAM PŘÍLOH

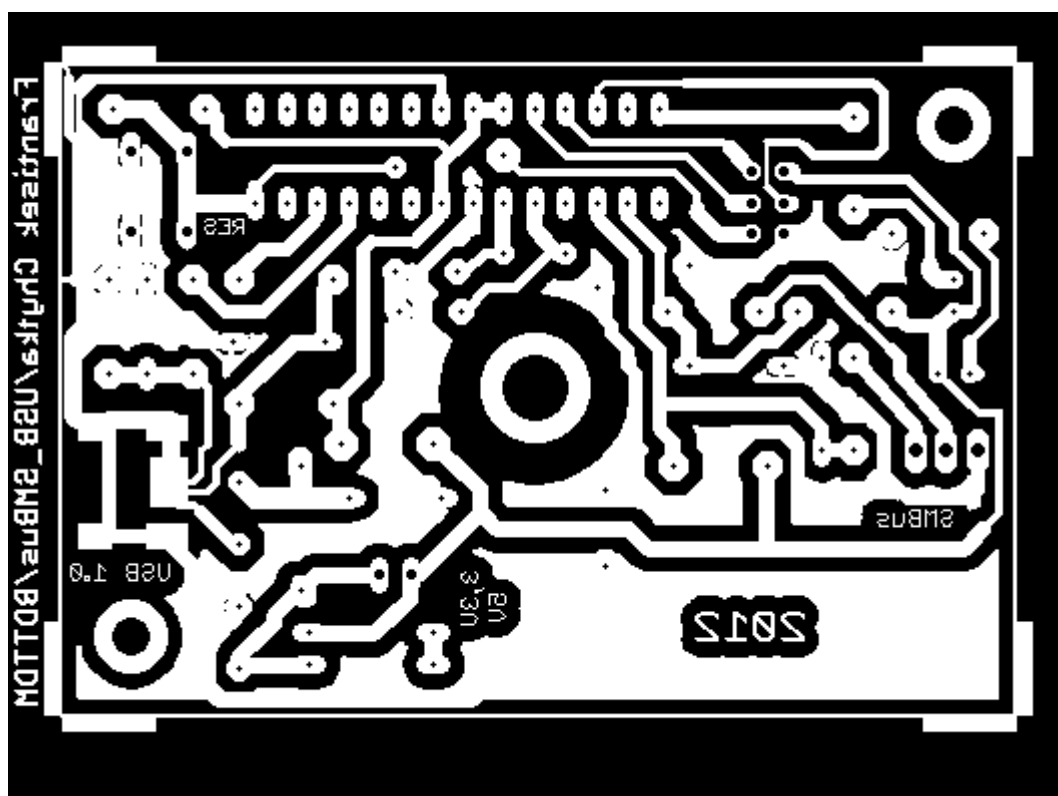
A	Hardware	45
A.1	Celkové schéma	45
A.2	Rozložení součástek na DPS.....	46
A.3	Vrstva vodivých cest (BOTTOM)	46
A.4	Seznam součástek	47
B	Software	48
B.1	Formulář aplikace – obecný přehled.....	48
B.2	Formulář aplikace – rozšířené možnosti	49
C	Uživatelský manuál	50
C.1	Instalace	50
C.2	Základní pokyny k ovládání	50
C.3	Základní možnosti.....	51
C.4	Rozšířené možnosti.....	52
C.5	Popis hardwaru	53

A.2 Rozložení součástek na DPS



Rozměr desky 53×78 mm, měřítko M1,56:1

A.3 Vrstva vodičů cest (BOTTOM)



Rozměr desky 53×78 mm, měřítko M1,56:1

A.4 Seznam součástek

Označení	Součástka	Označení	Součástka
C1	22p	R4	68R
C2	22p	R5	1k5
C3	100n	R6	1k5
D1	1N4446	R7	1k5
D2	1N4446	R8	150R
JP1	jumper 2,54 mm	R9	150R
L1	10 μ H	R10	47R
LED1	LED 3 mm	R11	47R
LED2	LED 3 mm	RES	tlačítko
MEGA168	ATMega168	SMBUS	jumper 2,54 mm
POLY1	Vratná pojistka 200 mA	X1	MINI USB konektor
PROG_C	konektor 6 pinů	ZD1-5.6V	BZX83V005.6
Q1	krystal 12 MHz	ZD2-3.6V	BZX83V003.6
R1	4k7	ZD3-3.6V	BZX83V003.6
R2	68R	ZD4-5.6V	BZX83V005.6
R3	1M	ZD5-5.6V	BZX83V005.6

B SOFTWARE

B.1 Formulář aplikace – obecný přehled

The screenshot displays the 'Smart Battery Monitor' application window. The interface is divided into several sections:

- Top Bar:** Contains the application title 'Smart Battery Monitor' and menu options 'File' and 'Help'.
- Left Panel:** Includes a 'CheckSum' section with a checked 'CRC8' option, an 'Autodetect' checkbox, a 'Transfer' button, and status indicators for 'SMB status' and 'USB status', both showing green checkmarks.
- Main Content Area:**
 - Identification:** Displays the following details:
 - Serial No.: 4416
 - Device Name: DELL X409G8C1
 - Manufacturer: SMP-PL18
 - Chemistry: LION
 - Current State:** Displays the following details:
 - Charging Cycles: 94
 - Voltage: 12,25 V
 - Remaining Capacity: 10,73 Wh
 - Full Charge Cap.: 10,93 Wh
 - Temperature: 28 °C
 - Original State:** Displays the following details:
 - Designed Capacity: 41,07 Wh
 - Designed Voltage: 11,1 V
 - Date of Manufacture: DD MM YYYY
 - DD: 1, MM: 12, YYYY: 2008
- Bottom Panel:** Features a 'READ' button, a list of detected devices (1. 0., DELL X409G8C1; 2. 4416, SMP-PL18, DELL X409G8C1), and buttons for 'Delete Entry', 'Delete All', and 'Save list'.
- Footer:** Shows 'Found HID Device' and a 'Help: ---' button.

B.2 Formulář aplikace – rozšířené možnosti

The screenshot displays the 'Smart Battery Monitor' application window. The title bar includes the application name and standard Windows window controls. The menu bar contains 'File' and 'Help'. The 'Advanced' tab is selected, showing several configuration sections:

- CheckSum:** A checkbox labeled 'CRC8' is checked.
- Autodetect:** An unchecked checkbox.
- Transfer:** A button with a right-pointing arrow.
- SMB status:** A green status indicator.
- USB status:** A green status indicator.

The main area is divided into three sections:

- Address:** Contains a 'Detect' button, a text field with the value '11', and a 'Use' button.
- Transmitted word (Tx (16b)):** Contains three input fields for 'Command' (23), 'Byte H' (2), and 'Byte L' (8). Below these are radio buttons for 'Read' (selected) and 'Write'. A 'Send' button is located to the right.
- Received word (Rx (16b)):** Displays the received data: Header (23), Byte H (0), and Byte L (94).

At the bottom, there is a 'Read block' section with a 'Send' button and a text field containing '34'. To the right of this is a 'Received block' section with a text field showing '-76-73-79-78' and a 'Clear RX buffer' button.

The status bar at the bottom right shows 'Found HID Device' and a 'Help' button.

C UŽIVATELSKÝ MANUÁL

C.1 Instalace

Požadavky na systém:

OS: Windows XP a vyšší

.NET Framework: 3.5 a vyšší

USB: USB 1.1 a vyšší

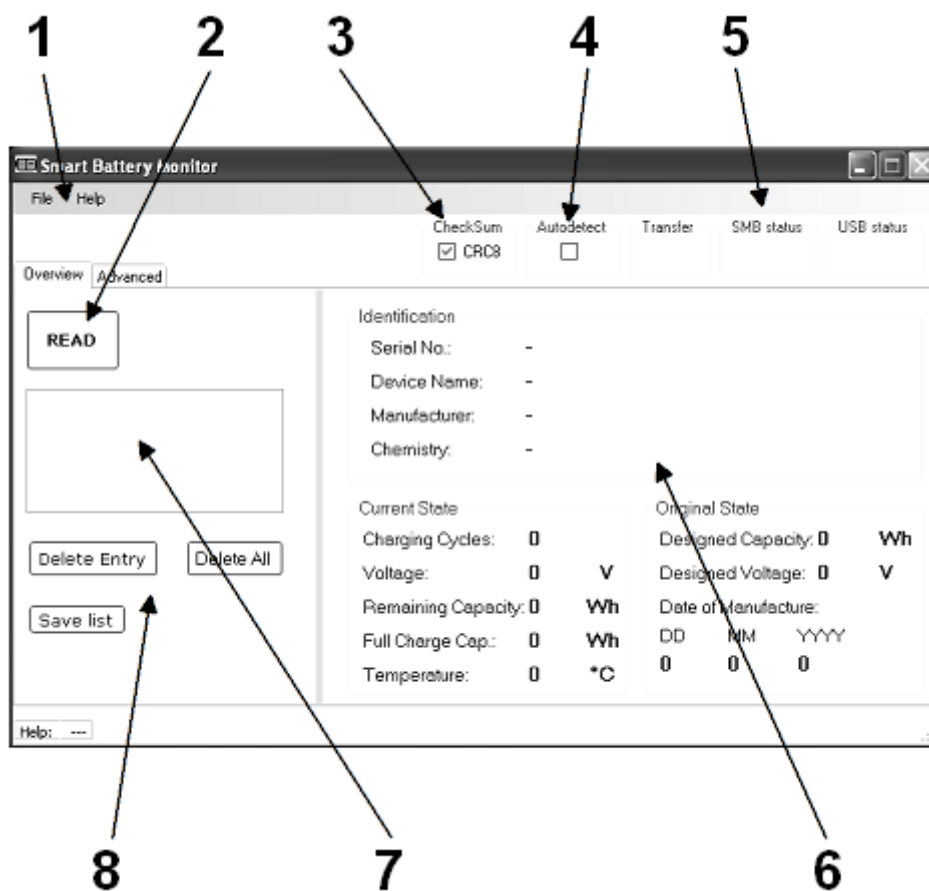
Instalace:

1. Spustit soubor „Setup.exe“.
2. Stisknout Spustit/Instalovat.
3. Je-li instalace zastavena antivirovým programem, je zapotřebí instalaci povolit a spustit znovu.

C.2 Základní pokyny k ovládání

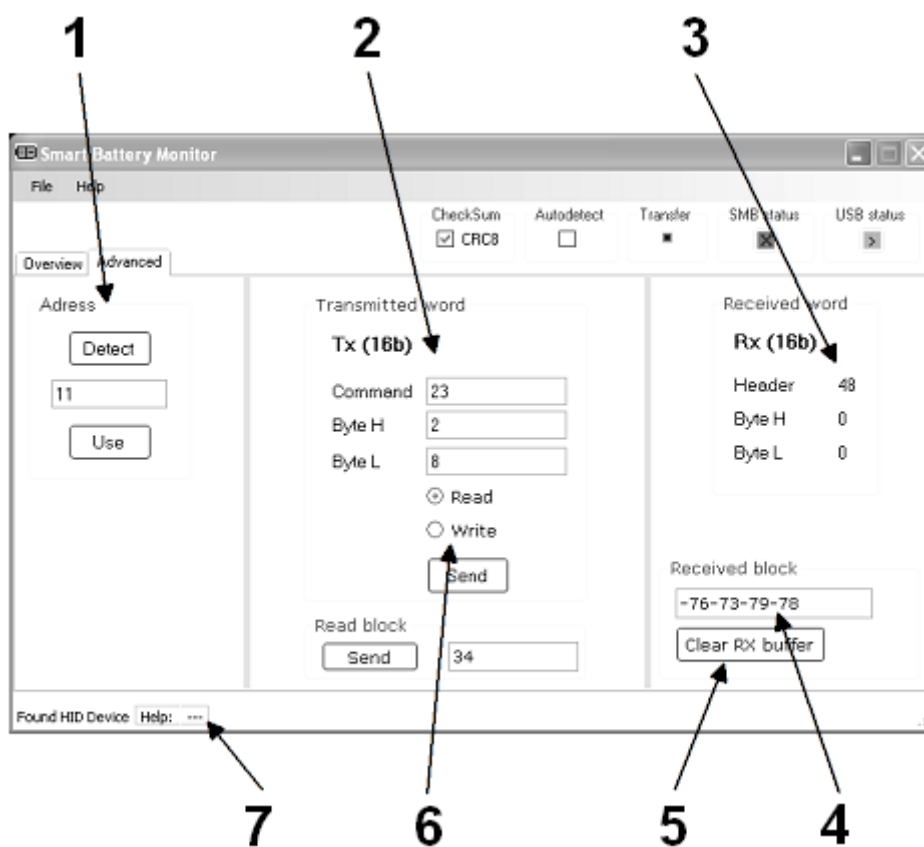
- Spustit program. Zobrazí se formulář s otevřenou záložkou „Základní možnosti“.
- Připojit zařízení do konektoru USB (pokud již není připojeno) a počkat na nalezení HW – signalizováno v poli „status USB“.
- Připojit sondu do konektoru SMBus.
- Spustit autodetekci akumulátoru.
- Nalézt na akumulátoru sběrnici SMBus. Projeví se to změnou statusu SMBus na formuláři – nalezeno zařízení je indikováno zelenou barvou
- Stisknout tlačítko „Read“ k načtení všech informací z akumulátoru.
- Stisknout tlačítko „Save list“ pro uložení informací na HDD.

C.3 Základní možnosti



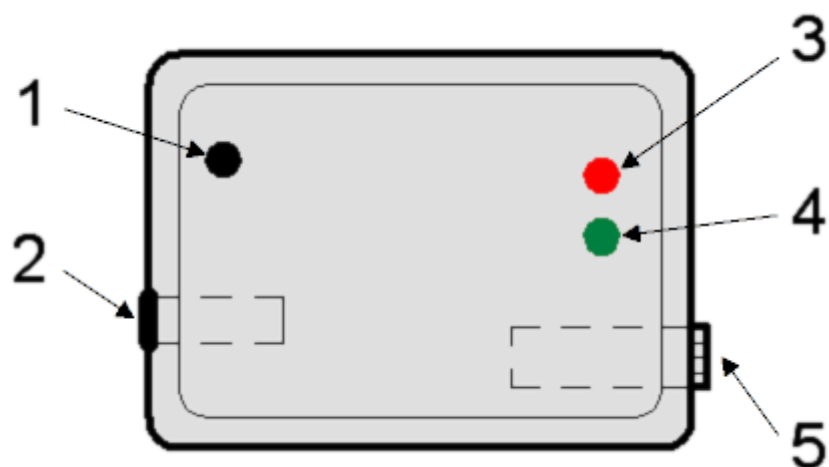
1. Menu (uložit seznam, ukončit program, nápověda),
2. načíst informace z akumulátoru, uložit do seznamu,
3. povolit/zakázat kontrolní součet,
4. spustit/vypnout automatickou detekci přítomnosti akumulátoru na SMBus,
5. indikátory (přenos dat, stav SMBus, stav USB),
6. detailní rozpis údajů o načteném akumulátoru,
7. seznam načtených akumulátorů,
8. ovládání seznamu akumulátorů (uložit, smazat).

C.4 Rozšířené možnosti



1. Spravovat adresy Smart Battery (detekce/nastavení),
2. sestavit uživatelský příkaz 16 bitů,
3. příchozí data 16 bitů,
4. příchozí blok dat,
5. vymazat zásobník příchozích bloků dat,
6. nastavit směr příkazu (čtení/zápis),
7. nápověda (popis při najetí myši).

C.5 Popis hardwaru



Pohled shora.

1. Tlačítko – reset
2. Konektor – USB 1.1
3. LED – Aktivita
4. LED – Napájení
5. Konektor – SMBus